Unequal Erasure Protection Technique for Scalable Multi-streams

Sorina Dumitrescu, *Member, IEEE,* Geoffrey Rivers, *Student Member, IEEE,* and Shahram Shirani, *Senior Member, IEEE*

Abstract

This paper presents a novel unequal erasure protection (UEP) strategy for the transmission of scalable data, formed by interleaving independently decodable and scalable streams, over packet erasure networks. The technique, termed multi-stream UEP (M-UEP), differs from the traditional UEP strategy by: 1) placing separate streams in separate packets to establish independence and 2) using permuted systematic Reed-Solomon codes to enhance the distribution of message symbols amongst the packets. M-UEP improves upon UEP by ensuring that all received source symbols are decoded. The R-D optimal redundancy allocation problem for M-UEP is formulated and its globally optimal solution is shown to have a time complexity of $O(2^N N(L+1)^{N+1})$, where N is the number of packets and L is the packet length. To address the high complexity of the globally optimal solution, an efficient sub-optimal algorithm is proposed which runs in $O(N^2L^2)$ time. The proposed M-UEP algorithm is applied on SPIHT coded images in conjunction with an appropriate grouping of wavelet coefficients into streams. The experimental results reveal that M-UEP consistently outperforms the traditional UEP reaching peak improvements of 0.6 dB. Moreover, our tests show that M-UEP is more robust than UEP in adverse channel conditions.

Index Terms

unequal erasure protection, scalable multi-streams, Reed-Solomon code, SPIHT.

©2008 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

An abstract of this work appeared in the Proceedings of the 2008 International Workshop on Multimedia Signal Processing, Cairns, Australia.

I. INTRODUCTION

Scalable image compression algorithms such as SPIHT [1] and EBCOT [2] provide a means for efficient representation of images in a successively refinable manner. That is, any prefix of the bit stream can reconstruct the image to a certain fidelity and the fidelity increases in the length of the decoded prefix. Moreover, a bit in a scalable bit stream can only be decoded if all previous bits have been decoded. Thus, the performance of scalable coders is highly susceptible to bit errors, since a single bit error can render the remainder of the bit stream undecodable.

During the transmission of information over packet switching networks, packets are randomly dropped when the network becomes over congested. The most typical approach for recovery from data loss is for the receiver to request retransmission of lost packets. However, this approach incurs additional delay which might not be acceptable for applications with strict time constraints. A more attractive approach in such cases is to apply forward error-correction (FEC) which protects against packet erasures by transmitting additional redundant data that facilitates the ability to reconstruct lost information at the receiver (e.g., Reed-Solomon codes).

Since the bits of a scalable bit-stream do not have equal importance, but rather their importance is related to their location in the bit-stream, unequal erasure protection (UEP) is the most suitable method of FEC. UEP employs a collection of *strict* systematic Reed-Solomon (RS) block codes (i.e. where the source symbols are grouped at the beginning of the codeword) of the same length but decreasing strength, to protect subsequent segments of the source bit-stream. In order to maximize the UEP performance, the erasure protection allocation can be optimized in rate distortion (R-D) sense. Multiple solutions to the UEP optimal redundancy assignment problem have been proposed in [3]-[10]. The packets to be sent through the lossy network are formed across the channel codewords. Any set of received packets can be used to reconstruct a prefix of the source stream which is decodable. Moreover, the fidelity of the reconstruction increases proportionally to the number of received packets. However, source symbols available at the decoder which do not belong to the reconstructed prefix cannot be decoded.

A key property of scalable coders such as SPIHT and EBCOT is that they produce bit-streams which are an interleaving of indepedently decodable and scalable streams. We will refer to such bit-streams as *multi-streams*. Researchers have exploited the multi-stream characteristic to achieve improved error-resilience to packet erasures. Creusere [12] employed multi-streams for Shapiro's embedded zerotree algorithm [13] by dividing the wavelet coefficients into groups and encoding each group independently. The individual bit-streams were then interleaved to form the multi-stream. This technique improves robustness since a bit

error only affects the stream to which it belongs, while all other streams can be fully decoded. Cho and Pearlman [14] extended the concept of independently decodable streams to scalable video (3D-SPIHT) and proposed equal error protection (EEP) using RCPC-CRC codes to improve robustness against channel errors. Alatan *et al.* [15] presented a UEP method for SPIHT encoded bit-streams which generates three streams which are protected by varying strength RCPC/CRC codes. Kim *et al.* [16] also proposed a UEP method for transmission of video multi-streams over the internet. They generated the streams by frequency domain partitioning of the wavelet coefficients, and determined the UEP protection for each stream independently.

Another method to achieve robustness was proposed by Rogers and Cosman in [17]. Error resilience was improved by grouping the streams into packets of fixed length. This method, which was tailored to transmission over packet erasure networks, ensured that all packets were independently decodable, and thus errors could not propagate beyond the packet in which they occurred. Wu *et al.* [18] further investigated rate-distortion optimal packetization with alignment constraints.

In this paper, in order to improve upon the traditional UEP technique, we also exploit the multi-stream characteristic of scalable code streams, while applying the concept of independently decodable packets to ensure that all received source symbols can be decoded. Packet independence is accomplished by assigning source symbols from a particular stream to a single packet. Additionally, we propose the use of *permuted* systematic RS codes (i.e., where the source symbols are interleaved with the redundancy symbols to form the codeword) instead of strict systematic codes, in order to maximize the flexibility of the erasure protection allocation. We will refer to the proposed technique as multi-stream UEP (M-UEP).

Note that transmission of separate streams in separate packets, in conjunction with UEP based on systematic RS codes has been proposed by Thomos *et al.* in [19]. However, the authors of [19] maintain the constraint of RS codes to be strict systematic. M-UEP eliminates this restriction, and thus allows for more flexibility in the erasure protection allocation. The same authors also present a product code technique (LDPC and RS codes) in [20], which employs symbol interleaving but does not consider independently decodable packets. In their scheme, random symbol interleaving is performed within each packet to randomize the errors and improve the efficiency of the LDPC decoding.

We formulate the R-D optimal redundancy allocation (RD-ORA) problem for M-UEP as a constrained integer non-linear optimization problem. We evaluate its time complexity to be $O(2^N N(L+1)^{N+1})$, where N is the number of packets and L is the packet length. This is done by converting the problem to a maximum-weight path problem in a weighted directed acyclic graph. Since the globally optimal solution is intractable, we propose a faster, sub-optimal solution algorithm, which runs in $O(N^2L^2)$ time. Our tests

on several SPIHT coded images show that the proposed M-UEP algorithm (with appropriate grouping of coefficients to obtain the streams), outperforms traditional UEP, reaching peak improvements of 0.6 dB. Moreover, in our experiments the advantage of M-UEP over UEP increases when channel mismatch occurs, fact which demonstrates the improved robustness of M-UEP in variable channel conditions.

The paper is structured as follows. Section II reviews the traditional UEP and introduces the proposed M-UEP strategy Section III formulates the R-D optimal redundancy allocation problem and discusses the globally optimal solution. An efficient sub-optimal algorithm is proposed in Section IV. Section V analyzes the overhead due to the side information needed at the decoder. An experimental comparison between the two UEP techniques is provided in Section VI and Section VII concludes the paper.

II. UEP STRATEGIES

This section begins with a review of the traditional UEP strategy and then introduces the proposed M-UEP technique. Both schemes are described in the context of a scalable multi-stream. We assume that the multi-stream is obtained by interleaving N independently decodable and scalable streams. For both strategies, we consider transmission over a lossy network, using N packets, of L symbols each, where a symbol is a sequence of a fixed number of bits (e.g., 8 bits). This structure can be envisioned as an array of $L \times N$ symbols which we will refer to as a *packetization array*.

A. Traditional UEP

In traditional UEP, a prefix of the scalable source sequence is partitioned into L consecutive segments of non-decreasing lengths, i.e., $1 \leq m_1 \leq m_2 \leq \ldots \leq m_L \leq N$, where m_i denotes the number of source symbols in segment i. The i-th segment is protected by a *strict* systematic (N, m_i) RS code. We use the term *strict* systematic to emphasize the fact that all the source symbols are placed at the beginning of the channel codeword. The key property of an (N, k) RS code is that, if at least k channel symbols are received at the decoder, then all lost symbols can be recovered. The i-th channel codeword consisting of m_i source symbols, followed by $N - m_i$ redundancy symbols fills the i-th row in the packetization array. Each column of the packetization array forms a packet, and all rows with RS channel codewords of equivalent strength form a layer. More formally, layer j ($1 \leq j \leq N$) is the set of rows with (N, j) RS channel codewords. Figure 1 illustrates a UEP packetization array for $N = 4$, $L = 8$. In this example, all layers are allocated a non-zero number of rows, however this is not a requirement.

When k packets of the N transmitted packets $(k \leq N)$ are received at the destination, all source symbols in the first k layers can be recovered due to the erasure protection. Since they form a prefix of

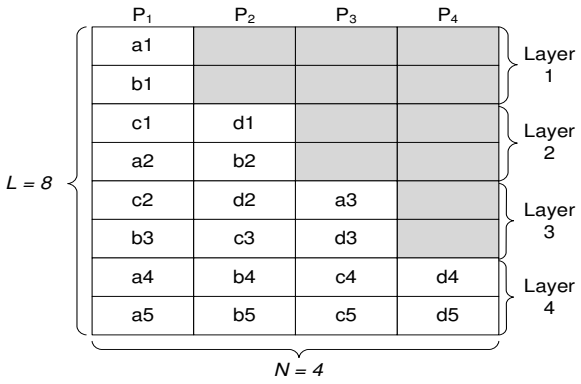

Fig. 1. **UEP packetization array:** $(N = 4, L = 8)$. Gray boxes represent redundancy symbols and white boxes source symbols.

the scalable code stream, the source data can be reconstructed to a certain fidelity. However, the source symbols in the received packets, that do not occur in the first k layers cannot be decoded, and must be discarded at the receiver. This is because such source symbols are separated from the reconstructed prefix by "holes" due to missing source symbols (i.e., those source symbols in layers $k+1$ through N, from the lost packets). We emphasize this disadvantage of UEP by means of an example.

Example 1: Consider the source multi-stream ($a1$, $b1$, $c1$, $d1$, $a2$, $b2$, $c2$, $d2$, $a3$, $b3$, $c3$, $d3$, $a4$, $b4$, $c4$, $d4$, $a5$, $b5$, $c5$, $d5$, $\cdots$), formed by interleaving four scalable streams A, B, C, D, where $x1, x2, x3, \cdots$, denote the symbols of stream X, $X \in \{A, B, C, D\}$. Consider now the UEP packetization with $(m_1, m_2, m_3, m_4, m_5, m_6, m_7, m_8) = (1, 1, 2, 2, 3, 3, 4, 4)$, which is illustrated in Figure 1. Let us analyze the case when only packets P_2 and P_4 arrive at the receiver. Due to the erasure protection, only the lost symbols in layers 1 and 2 can be recovered, i.e., $a1$, $b1$, $c1$, $a2$. Together with the source symbols from the received packets they form the prefix $(a1, a2)$ of stream A, prefix $(b1, b2)$ of B, prefix $(c1)$ of C and prefix $(d1, d2)$ of D, which are decoded. Note that the symbols $c3, b4, b5, d4, d5$ are also available at the decoder, but they cannot be decoded ($c3$ cannot be decoded because $c2$ is missing, $b4, b5$ cannot be decoded because $b3$ is missing, etc.).

B. Multi-stream UEP (M-UEP)

In the M-UEP strategy the source symbols in packet i constitute a prefix of stream i. Each row is formed by a *permuted* systematic RS channel codeword. A *permuted* systematic RS codeword is obtained from a *strict* systematic RS codeword, by applying a permutation to the channel symbols. This causes the source symbols to be interleaved with the redundancy symbols in the channel codeword. Clearly, the

erasure protection capabilities are not affected. In other words, an (N, k) *permuted* systematic RS code is able to correct up to $N - k$ erasures. As with UEP, the strengths of the RS channel codewords in M-UEP must be non-increasing as the row index increases.

Let x_j denote the number of rows in layer j, and let $x_j^{(i)}$ be the number of source symbols from substream i (packet i), situated in layer j, $1 \leq i, j \leq N$. Note that the rows belonging to any layer are consecutive, due to the constraint which imposes non-increasing strength of the RS codes in the row index. As in UEP, some layers j may be empty, hence with $x_j = 0$. Figure 2 illustrates an M-UEP packetization array with $N = 4$, $L = 8$, for the same multi-stream as in Example 1.

When only k packets of the N transmitted packets $(k \leq N)$ are received at the destination, all source symbols in the first k layers from any lost packet i can be recovered due to the erasure protection as in the case of UEP. These symbols form a prefix of stream i, therefore they can be decoded. Moreover, all source symbols from any received packet ℓ can also be decoded since they form the transmitted prefix of stream ℓ. Thus, as in UEP, all source symbols in the first k layers will be recovered and decoded. Moreover, additionally, all available source symbols in layers $k+1$ through N from the received packets, will also be decoded. This key benefit of M-UEP over UEP is illustrated by the following example.

Example 2: Consider the same source multi-stream as Example 1. Figure 2 illustrates an M-UEP packetization array where the streams A, B, C, D are assigned to packets P_1, P_2, P_3, P_4, respectively. Furthermore, each source symbol has the same erasure protection as in the UEP scenario of Example 1. Let us revisit the case when only packets P_2 and P_4 are received. In the context of M-UEP, source symbols in the first two layers from the lost packets ($a1$, $a2$, $c1$) can be recovered. Thus, prefixes $(a1, a2)$ of stream A and $(c1)$ of C are formed and decoded. Moreover, the source symbols from the received packets form prefixes $(b1, b2, b3, b4, b5)$ of B and $(d1, d2, d3, d4, d5)$ of D which are also decoded. Thus, the reconstruction is strictly better than in the UEP case.

The advantage of M-UEP over UEP is clear in Example 2. Notice that in this example the redundancy assignment to source symbols is identical to the redundancy assignment achieved under the UEP framework. When this happens, in other words when any M-UEP layer j contains the same source symbols as the UEP layer j, the M-UEP technique ensures a performance improvement in any situation when only a subset of $k, k < N$ packets is received. This is because the M-UEP guarantees the decoding of all source symbols that would have been decoded under the UEP scenario (i.e., all source symbols in layers 1 through k), and additionally decodes all source symbols in layers $k + 1$ through N of the received packets.

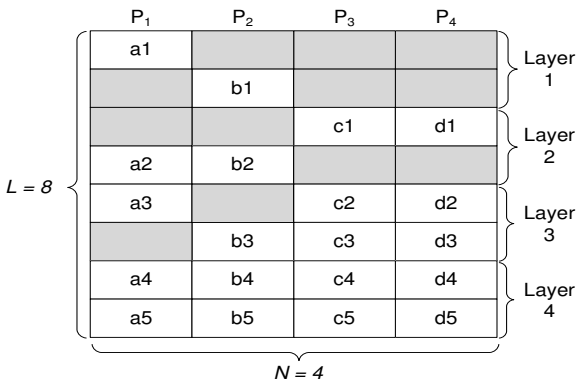

Fig. 2. **M-UEP packetization array:** $(N = 4, L = 8)$. Gray boxes represent redundancy symbols and white boxes source symbols.

The ability of the M-UEP framework to ensure the same redundancy allocation as the UEP is influenced by the nature of the streams. A sufficient condition for this property to hold is the multi-stream to be balanced. A multi-stream is balanced when symbols of the same importance in R-D sense are evenly distributed amongst the streams. Assuming R-D optimal interleaving of streams in the multi-stream, a balanced multi-stream ensures that each UEP layer j contains approximately equal number of symbols from each stream. If x_j denotes the number of rows in UEP layer j, then approximately jx_j/N source symbols from each stream are contained in this layer. Since $jx_j/N \leq x_j$, these source symbols can be arranged to fill x_j rows of layer j in the M-UEP packetization array, such that each column contains only symbols from the corresponding stream (a rigorous proof of this claim is provided in Appendix A).

When the multi-stream is unbalanced, the constraint of assigning source symbols from stream i only to packet i may prevent M-UEP from achieving identical redundancy protection for each source symbol as with UEP. However in the case of a mildly unbalanced multi-stream, M-UEP can still outperform UEP, especially under poor channel conditions. This is due to the fact that when only a small number of packets are received, the number of additional symbols decoded in the M-UEP framework is large enough to compensate for the unavailability of a few important source symbols which might have been available to the decoder in the UEP case. On the other hand, if the multi-stream is severely unbalanced, some important symbols might not even be transmitted with M-UEP since they do not fit into the corresponding packet. This loss might offset the contribution of the less important additional symbols decoded. However, it is difficult to provide a precise characterization of the degree of imbalance which deprives the M-UEP of its advantage over UEP.

III. R-D Optimal Redundancy Allocation

In Example 2 from the previous section, the M-UEP layers have the same size as the UEP layers. However, such an allocation of redundancy is not necessarily optimal for M-UEP. To maximize the M-UEP performance, we address the problem of optimal redundancy allocation (ORA) in the rate-distortion (R-D) sense, formulated as follows: find the redundancy assignment which minimizes the expected distortion at the receiver, given the total transmission budget of $N \times L$ symbols, the channel loss statistics, and the R-D curves of the streams.

Let us denote by $P_N(k)$ the probability that out of the N transmitted packets, k are lost, while $N-k$ are received. We assume that the values $P_N(k)$ are known at the encoder. Furthermore, we assume that the distortion metric is additive, in other words, each decoded symbol contributes additively to the total distortion reduction. This assumption holds true when the mean-squared error (MSE) is used as the distortion metric and no error concealment is applied at the decoder. Formally, for each stream i, let $\Delta D_i(r_i)$ denote the decrease in distortion due to decoding the r_i-th source symbol from stream i. For convenience, we also use the notation $\Delta D(r)$ for the decrease in distortion due to decoding the r-th source symbol from the multi-stream. Notice that if the r-th symbol in the multi-stream coincides with the r_i-th symbol from stream i, then $\Delta D(r) = \Delta D_i(r_i)$.

Next we review the RD-ORA problem for the UEP framework.

A. UEP RD-ORA

For UEP-based transmission, the probability $C_U(j)$ that a source symbol in layer j will be decoded equals the probability that at least j packets are received. Consequently,

$$C_U(j) = \sum_{k=0}^{N-j} P_N(k)\,, 1 \le j \le N. \tag{1}$$

Let x_j denote the size of layer j (i.e., the number of rows allocated to layer j), $1 \le j \le N$. Then the number of source symbols in any layer ℓ is ℓx_ℓ. Consequently, the symbols situated in some layer j are all the symbols in the multi-stream between positions $(\sum_{\ell=1}^{j-1} \ell x_\ell) + 1$ and $\sum_{\ell=1}^{j} \ell x_\ell$, inclusive. Then the expected distortion of the reconstructed source at the receiver is

$$\bar{D}_U = D_{max} - \sum_{j=1}^{N} \left(\sum_{r=1+\sum_{\ell=1}^{j-1} \ell x_\ell}^{\sum_{\ell=1}^{j} \ell x_\ell} \Delta D(r) \right) \times C_U(j), \tag{2}$$

where D_{max} denotes the distortion when no source symbols are decoded.

The objective of the UEP R-D ORA problem is to find the non-negative integers $x_1, x_2, \cdots, x_N$ which minimize $\bar{D}_U$ subject to the constraint $\sum_{j=1}^{N} x_j = L$. This problem has been extensively studied by the

research community [3]-[10]. A plethora of solutions of various degrees of accuracy and/or efficiency have been proposed. The fastest globally optimal algorithm for the most general case, presented in [8], has $O(N^2L^2)$ time complexity. A detailed discussion and comparison of notable solutions to the RD-ORA UEP problem can be found in [11].

B. M-UEP RD-ORA

In order to formulate the M-UEP RD-ORA problem, recall that in M-UEP-based transmission, a source symbol of stream i, situated in layer j, will be recovered and decoded at the receiver if and only if one of the following events occurs: 1) packet i is received at the destination, or 2) packet i is lost and at least j other packets are received. Let us denote the probability of the first event by $\alpha(i)$, and the probability of the second event by $\beta(i,j)$. Consequently, the probability that a source symbol of stream i, situated in layer j, will be decoded is $C_M(i,j) = \alpha(i) + \beta(i,j)$, for $1 \leq i,j \leq N$. Recall that $x_j^{(i)}$ denotes the number of source symbols from stream i situated in layer j, and x_j denotes the number of rows in layer j, $1 \leq i,j \leq N$. Because the distortion is additive, and the expectation operator is a linear operator, it follows that the expected distortion of the reconstructed source at the receiver, denoted by $\bar{D}_M$, can be expressed as the maximum distortion when no source symbols are decoded (D_{max}) reduced by the contribution of each symbol to the decrease of distortion, weighted by the probability that the symbol is decoded. Thus, we obtain

$$\bar{D}_M = D_{max} - \sum_{j=1}^{N}\sum_{i=1}^{N} C_M(i,j) \left(\sum_{r_i=1+\sum_{\ell=1}^{j-1} x_\ell^{(i)}}^{\sum_{\ell=1}^{j} x_\ell^{(i)}} \Delta D_i(r_i) \right). \tag{3}$$

The objective of the M-UEP RD-ORA problem is to minimize $\bar{D}_M$ over all non-negative integers $x_j, x_j^{(i)}$, $1 \leq i,j \leq N$, satisfying the constraints

$$\sum_{j=1}^{N} x_j = L \tag{4}$$

$$\sum_{i=1}^{N} x_j^{(i)} = j x_j \ \forall\, j,\ 1 \leq j \leq N, \tag{5}$$

$$x_j^{(i)} \leq x_j \ \forall\, i,j,\ 1 \leq i,j \leq N. \tag{6}$$

It is easy to see that the above constraints are necessary. Constraints (5) follow from the fact that any row of layer j must have exactly j source symbols. Constraints (6) are due to the fact that symbols from stream i can only be placed in packet i. These constraints are also sufficient. We defer the proof of their sufficiency to Appendix A in order to avoid interrupting the flow of the exposition.

In Appendix B, we show that the M-UEP RD-ORA problem can be cast as a maximum-weight path problem in a weighted directed acyclic graph, yielding a globally optimal solution algorithm of $O(2^N N(L+1)^{N+1})$ time complexity. This is a tight bound and holds in both worst and best cases. Note that this complexity value is polynomial in L, but exponential in N. Therefore, we conclude that for all but small values of N, the globally optimal solution is impractical. This fact motivates the use of a sub-optimal algorithm to solve the problem, which is the topic of the next section.

IV. Sub-optimal Solution to M-UEP RD-ORA Problem for Symmetric Channels

In this section we develop a fast sub-optimal solution algorithm for the M-UEP RD-ORA problem, in the case of symmetric packet loss channels.

Definition 1. A packet loss channel is said to be *symmetric* if the probability that the packets in some subset $\mathcal{I} \subset \{1, 2, \cdots, N\}$ are lost, while the rest of packets are received, is the same for all subsets $\mathcal{I}$ of equal size.

As proved in Appendix C, for symmetric channels the coefficient $C_M(i, j)$ does not depend on i, thus it can be denoted by $C_M(j)$, and we have

$$C_M(j) = 1 - \mu + \sum_{k=0}^{N-j} \frac{k}{N} P_N(k), \tag{7}$$

where μ denotes the mean packet loss rate. Thus, the cost function (3) becomes

$$\bar{D}_M = D_{max} - \sum_{j=1}^{N} \left(C_M(j) \sum_{i=1}^{N} \left(\sum_{r_i=1+\sum_{\ell=1}^{j-1} x_\ell^{(i)}}^{\sum_{\ell=1}^{j} x_\ell^{(i)}} \Delta D_i(r_i) \right) \right). \tag{8}$$

The proposed algorithm hinges on the assumption that in the problem solution all source symbols situated in a layer form a contiguous segment of the multi-stream. As proved by the following lemma, this assumption holds true for multi-streams with convex R-D curve, if we disregard the constraint (6).

Lemma. Assume that the multi-stream has convex R-D curve (i.e., $\Delta D(r) \geq \Delta D(r+1)$ for any $r \geq 0$). Consider the problem of minimizing (8) subject to the constraints (4) and (5). Then there is an optimal solution such that, for any $j, 1 \leq j \leq N$, all source symbols situated in layer j form a contiguous segment of the multi-stream.

Proof. It is sufficient to show that, for each $j, 1 \leq j \leq N-1$, each symbol in layer j must contribute a distortion reduction at least equal to the distortion reduction of any symbol in layer $j+1$. To prove this claim, we consider the scenario when it is not satisfied. Thus, assume there are packets i_1 and i_2, source

symbol $\tilde{a}$ placed in layer j of packet i_1, and source symbol $\tilde{b}$ placed in layer $j+1$ of packet i_2, such that $\Delta(\tilde{b}) > \Delta(\tilde{a})$, where $\Delta(s)$ denotes the contribution of symbol s to the decrease in distortion. Further, let a be the last source symbol of packet i_1 in layer j and b be the first source symbol of packet i_2 in layer $j+1$. The convexity of the R-D curves of the streams implies that $\Delta(b) > \Delta(a)$. This scenario is illustrated in Figure 3(a) for $i_1 = N-1$ and $i_2 = 2$. Now let us promote source symbol b to layer j and demote source symbol a to layer $j+1$. By doing this, constraints (4), (5) are not broken (recall that we do not care of (6) for now), consequently this swap yields a feasible solution . This new solution is illustrated by Figure 3(b). By performing this source symbol swap between layers j and $j+1$, the cost function of (8) will decrease by the amount $(C_M(j) - C_M(j+1))(\Delta(b) - \Delta(a)) > 0$, thus contradicting the optimality of the previous solution. □

Note that if all source symbols situated in a layer form a contiguous segment of the multi-stream, then the cost function of (8) can be rewritten as

$$D_{max} - \sum_{j=1}^{N} C_M(j) \left(\sum_{r=1+\sum_{\ell=1}^{j-1} \ell x_\ell}^{\sum_{\ell=1}^{j} \ell x_\ell} \Delta D(r) \right). \tag{9}$$

Inspired by the above observations, we propose a sub-optimal algorithm for the M-UEP RD-ORA problem for symmetric channels, which proceeds in two main steps. Step 1 minimizes the objective function of (9) over all non-negative integers x_j satisfying (4). Step 2 uses the values x_j output at Step 1 and finds the values $x_j^{(i)}$ such that the constraints of (5) and (6) to be satisfied. Notice that the problem of Step 1 is similar to the UEP RD-ORA problem, the only difference consisting in the calculation of the weights $C_M(j)$. Therefore any algorithm which solves the UEP RD-ORA problem (and which does not rely on particular properties of the weights) can be used for this purpose. In our experiments we employ the $O(N^2L^2)$ time globally optimal solution developed in [8]. For Step 2 we use the algorithm whose pseudocode is provided in Figure **??**. The algorithm proceeds in increasing order of j. For each j, jx_j iterations are performed. At each iteration exactly one source symbol is assigned to layer j. The algorithm maintains pointers for each stream to the current candidate symbol to be placed in layer j (i.e., the first symbol unassigned a layer yet). Also a list $\mathcal{I}$ is maintained of streams for which the number of symbols already assigned to layer j, is smaller than x_j (hence streams in $\mathcal{I}$ have not yet fully occupied their designated capacity in layer j). The symbol to be assigned at each iteration is the symbol with highest distortion reduction among all candidate symbols from streams in list $\mathcal{I}$. Clearly, this algorithm ensures that the output satisfies conditions (5) and (6). The number of operations performed is $O(\sum_{j=1}^{N} jx_j) = O(NL)$. We conclude that completing steps 1 and 2 requires $O(N^2L^2)$ time.

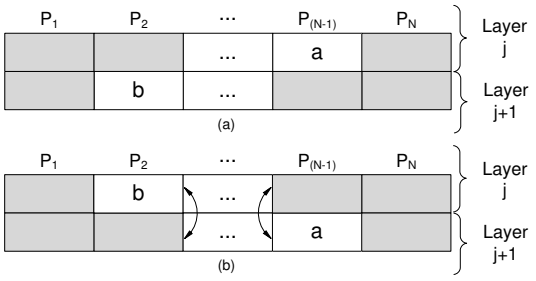

Fig. 3. **M-UEP Packetization Array Cross section :** M-UEP packetization array layers j and $j+1$ prior to swapping source symbols (a), and after swapping source symbols (b). Gray boxes represent redundancy symbols and white boxes represent source symbols.

It is instructive to identify cases when the proposed solution to the M-UEP RD ORA problem is optimal.

Definition 2. We say that the multi-stream is *perfectly balanced* if any segment of N consecutive symbols of the multi-streams contains exactly one symbol from each stream.

Proposition. Assume that the multi-stream is perfectly balanced and its R-D curve is convex. Then the algorithm proposed in this section guarantees the optimal solution to M-UEP RD ORA problem for symmetric channels.

Proof. By Lemma the output of Step 1 is the optimal solution to the problem of minimizing (8) subject to constraints (4) and (5). Consequently, the optimal value of the cost function in (9), denoted by $\bar{D}_{opt}$, is a lower bound for the minimum M-UEP expected distortion. Denote by $\bar{D}_{M,opt}$, the M-UEP expected distortion obtained by using the algorithm described in this section, and by U_j the segment of the multi-stream between positions $(\sum_{\ell=1}^{j-1} \ell x_\ell)+1$ and $\sum_{\ell=1}^{j} \ell x_\ell$, for the output of Step 1. Further, let $u_j^{(i)}$ denote the number of source symbols from stream i, situated in segment U_j, for all i, j. It is easy to see that if conditions

$$u_j^{(i)} \leq x_j \ \forall i, j, \ 1 \leq i, j \leq N, \tag{10}$$

are satisfied, then the algorithm of Step 2 ensures that $x_j^{(i)} = u_j^{(i)}$ for all i, j, which, in turn, implies that $\bar{D}_{M,opt} = \bar{D}_{opt}$. On the other hand, since the multi-stream is perfectly balanced, we have $u_j^{(i)} \leq \lceil j x_j / N \rceil$, where $\lceil \cdot \rceil$ denotes the ceiling function. Since x_j is an integer and $j x_j / N \leq x_j$, it further follows that $\lceil j x_j / N \rceil \leq x_j$. Consequently, relations (10) are satisfied, fact which concludes the proof. □

For a multi-stream with convex R-D curve for which the violations in relations (10) are mild, i.e.,

```
current (i) = index of current candidate symbol of stream i
I = set of stream indices i ∈ {1, 2, 3, ..., N} that satisfy
x_j^(i) < x_j
set current(i) = 0 and x_j^(i) = 0 , ∀ i, j = 1 to N
for j = 1 to N
  set I = {1, 2, 3, ..., N}
  for q = 1 to j x_j
    s = argmax_{k∈I} ΔD_k (current (k))
    increment x_j^(s) and current(s) by 1
    if x_j^(s) equals x_j
      remove s from I
    end if
  end for
end for
```

Fig. 4. **Source Symbol Assignment :** M-UEP packetization array source symbol assignment pseudocode.

the excess $u_j^{(i)} - x_j$ is small, we expect the solution to be close to optimal. An intuitive explanation is the following. Violations of relations (10) cause excess symbols which do not fit into the layer allocated according to Step 1, to be placed into subsequent layers, thus reducing their protection level. If $u_j^{(i)} - x_j$ is small, it is expected that the excess symbols will fit into the next layer. Since this leads to only a small change in the protection level (especially when N is high), it is likely that $\bar{D}_{M,opt}$ increases only slightly from the lower bound. Since $\bar{D}_{M,opt} - \bar{D}_{opt}$ is an upper bound for the gap between $\bar{D}_{M,opt}$ and the optimal solution, this guarantees only a small degradation from the optimum. On the other hand, if the violations in relations (10) are severe, Step 2 will cause high changes in protection level of source symbols, pushing $\bar{D}_{M,opt}$ far away from the lower bound. In such a case the proposed redundancy allocation algorithm does not have any performance guarantee.

One last thing worth mentioning before concluding this section is that $\bar{D}_{opt}$ is always lower than $\bar{D}_{U,opt}$, where $\bar{D}_{U,opt}$ denotes the minimum UEP expected distortion. This is due to the fact that $C_M(j) > C_U(j)$ for all $1 \leq j \leq N$ and both optimization problems have the same set of feasible solutions. Moreover, $\bar{D}_{opt}$ depends on the RD curve of the multi-streams, but does not depend on how the source symbols are distributed among the streams. Therefore, the difference $\bar{D}_{U,opt} - \bar{D}_{opt}$ is an upper bound on the possible advantage of M-UEP over UEP, which does not depend on the balance properties of the multi-stream.

V. M-UEP Side Information

A critical aspect of M-UEP that must be considered is the side information which specifies the M-UEP packetization to the decoder. In the formulation of the RD-ORA problem we have not accounted for this

side information, therefore it is important to analyze the overhead resulting from it. This side information can be split into two parts: SI_1 - the side information needed to specify the size of each layer (i.e., the values x_j), and SI_2 - the additional side information needed to completely determine the M-UEP packetization, given the knowledge of SI_1.

Since $0 \leq x_j \leq L$, it follows that $\lceil \log_2(L+1) \rceil$ bits are enough to encode the value of some x_j. Moreover, only $x_1, \cdots, x_{N-1}$ have to be specified since x_N can be computed from (4). In conclusion, $si_1 = (N-1)\lceil \log_2(L+1) \rceil$ bits are enough to encode SI_1.

In order to encode SI_2, for each non-empty layer $j, j \neq N$, we use $(N-1)\lceil \log_2(x_j+1) \rceil$ bits to indicate the values $x_j^{(i)}$, $1 \leq i \leq N-1$. Note that $x_j^{(N)}$ can be computed from (5). Thus, a total of $si_2 = (N-1)\sum_{j=1, x_j \neq 0}^{N-1} \lceil \log_2(x_j+1) \rceil$ bits suffice to encode SI_2. Clearly, this amount depends on the number of allocated layers, which, in turn, depends on the type of channel, as revealed by our experiments in Section VI. However, an upper bound for this value can be computed.

$$\begin{aligned} si_2 &= (N-1)\sum_{j=1}^{N-1} \lceil \log_2(x_j+1) \rceil \\ &\leq (N-1)\sum_{j=1}^{N-1} (\log_2(x_j+1)+1) \\ &= (N-1)\log_2\left(\Pi_{j=1}^{N-1}(x_j+1)\right)(N-1)^2 \\ &\leq (N-1)\log_2\Big(\sum_{j=1}^{N-1}(x_j+1)/(N-1)\Big)^{N-1}(N-1)^2 \\ &\leq (N-1)^2\left(\log_2(L/(N-1)+1)+1\right). \end{aligned} \tag{11}$$

The second inequality in the above sequence of relations follows from the inequality between the geometric and arithmetic means of $N-1$ positive numbers. Finally, an upper bound for the total amount of side information is

$$\begin{aligned} si_1 + si_2 \leq (N-1)(\log_2(L+1)+1) + \\ (N-1)^2\left(\log_2(L/(N-1)+1)+1\right). \end{aligned} \tag{12}$$

VI. Experimental Results

The main goal of this section is to compare the performance of the proposed M-UEP strategy versus traditional UEP. For this we performed tests on two 512×512 images (Lena and Peppers). In both cases a 5-level Cohen-Daubechies-Feauveau 9/7 wavelet transform was applied, and the resulting wavelet coefficient matrix was SPIHT encoded (without arithmetic coding). In essence, the code stream produced

by SPIHT can be regarded as the interleaving of 256 independent *primary* streams, with 64 streams corresponding to non-root nodes in the lowest sub-band, and the remaining 192 streams representing spatial orientation trees. In our experiments, we considered $N = 2, 4, 6, \ldots, 40$. In order to obtain N independent streams, we group the primary streams into N groups. The primary streams of each group are then interleaved to form a stream, such that the relative order of symbols in each stream is the same as in the SPIHT coded multi-stream. As emphasized throughout the paper, the M-UEP performance is affected by the balance properties of the multi-stream. Therefore, it is expected that the grouping technique will influence the M-UEP performance.

In our experiments we use two grouping strategies: 1) predefined and 2) optimized grouping. In order to reduce the amount of side information needed to specify the grouping, we first fix an ordering of the primary streams, and constrain each group to be formed by consecutive primary streams according to this ordering. In predefined grouping, each group contains the same number of primary streams, i.e. P/N, where P is the total number of primary streams. In this case no side information is needed to indicate the grouping. In optimized grouping, the number of primary streams assigned to each group is optimized using the criterion in [18]. Once the streams are formed, a header of $2\lceil \log_2 P \rceil$ bits is added to each stream indicating the first and the last primary stream in the corresponding group. In optimized grouping the number of primary streams in each group is decided such that the distortion achieved when all the prefixes of size L from all formed streams are decoded, to be minimized. A similar problem, with slightly different constraints, was formulated in [18] and solved by using dynamic programming. The dynamic programming algorithm to solve our problem can be easily derived from [18]. Due to space limitation we omit the description of the algorithm here.

For the fixed ordering of primary streams we have considered two examples. One is the dispersed dot-dithering ordering (DD) [17], [18] borrowed from digital halftoning. The other ordering, which we refer to as subband dispersed (SD), is inspired by the DD ordering, but achieves a better dispersion of primary streams corresponding to non-root nodes. To define the SD ordering, consider, for $k \leq 1$, the $2^k \times 2^k$-dimensional matrices MA_k, MB_k, MC_k, MD_k defined recursively as follows

$$MA_1 = \begin{bmatrix} 0 & 4 \\ 8 & 12 \end{bmatrix}, \; MB_1 = \begin{bmatrix} 13 & 1 \\ 5 & 9 \end{bmatrix},$$

$$MC_1 = \begin{bmatrix} 10 & 14 \\ 2 & 6 \end{bmatrix}, \; MD_1 = \begin{bmatrix} 7 & 11 \\ 15 & 3 \end{bmatrix}.$$

$$X_{k+1} = \begin{bmatrix} X_k & X_k + 1 \times 2^{2k+2} \times U_k \\ X_k + 2 \times 2^{2k+2} \times U_k & X_k + 3 \times 2^{2k+2} \times U_k \end{bmatrix},$$

for $X \in \{MA, MB, MC, MD\}$, $k \leq 1$, where U_k denotes the $2^k \times 2^k$-dimensional matrix whose all elements are equal to 1. Finally, for $k \geq 2$, define the $2^k \times 2^k$-dimensional matrix M_k as

$$M_k = \begin{bmatrix} MA_{k-1} & MB_{k-1} \\ MC_{k-1} & MD_{k-1} \end{bmatrix}.$$

If the number of rows and columns in the lowest subband is 2^m, then the matrix M_m defines the index in the SD ordering corresponding to each primary stream. Note that the indexing starts at 0. Thus, when $P = 64$, hence $m = 3$, the assignment matrix for SD ordering is

$$M_3 = \begin{bmatrix} 0 & 4 & 16 & 20 & 13 & 1 & 29 & 17 \\ 8 & 12 & 24 & 28 & 5 & 9 & 21 & 25 \\ 32 & 36 & 48 & 52 & 45 & 33 & 61 & 49 \\ 40 & 44 & 56 & 60 & 37 & 41 & 53 & 57 \\ 10 & 14 & 26 & 30 & 7 & 11 & 23 & 27 \\ 2 & 6 & 18 & 22 & 15 & 3 & 31 & 19 \\ 42 & 46 & 58 & 62 & 39 & 43 & 55 & 59 \\ 34 & 38 & 50 & 54 & 47 & 35 & 63 & 51 \end{bmatrix}.$$

Thus, by applying the two grouping strategies for each of the two orderings (DD and SD) we obtained four groupings which were tested in our experiments. For brevity we will use the acronyms PDD and PSD to refer to the predefined grouping based on DD, SD, respectively, and the acronyms OptDD and OptSD for the optimized grouping counterparts.

We implemented the M-UEP transmission scheme for each of the four groupings introduced above. In each case we used the algorithm of Section IV to solve the M-UEP RD-ORA problem. For Step 1 we employed the globally optimal algorithm for UEP of [8], which does not require the convexity assumption. The same algorithm of [8] was used to solve UEP RD-ORA for the SPIHT codestream without arithmetic coding Furthermore, for both M-UEP and UEP we assumed that the side information needed to specify the packetization was sent to the decoder via a secure channel. However, the grouping side information, when needed, was included in the M-UEP packets.

Our tests were performed for two symmetric packet loss channel models. The first one is the independent packet loss (IPL) model with erasure rates $\epsilon = 0.05, 0.15$. Independence of packet losses is a reasonable assumption when the packets are interleaved with packets from other applications during

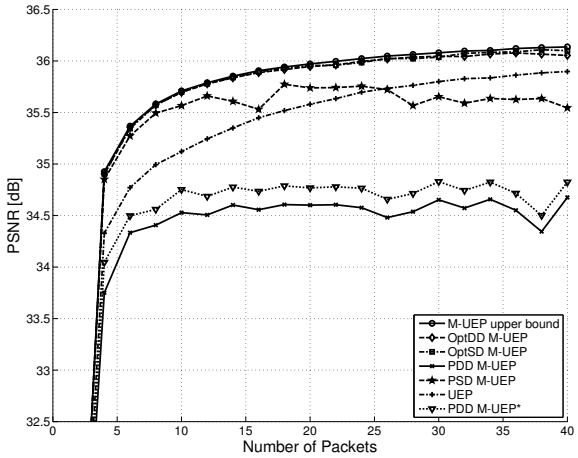

Fig. 5. **Grouping comparison:** M-UEP PSNR vs. number of packets, for at $R = 0.50$ bpp, for IPL $\epsilon = 0.05$.

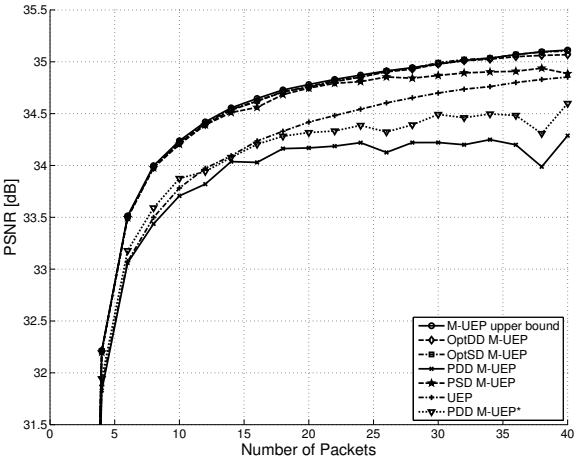

Fig. 6. **Grouping comparison:** M-UEP PSNR vs. number of packets, for at $R = 0.50$ bpp, for IPL $\epsilon = 0.15$.

the transmition. The second model is a channel with exponentially decreasing probability mass function $P_N(k), 1 \leq k \leq N$, (EPL) with mean packet loss values $\mu = 0.05, 0.15$.

For each value of $N = 2, 4, 6, \cdots, 40$, we have considered two values for the packet lengths L, such that the total transmission rate (denoted by R) to be 0.20 and 0.50 bits per pixel (bpp), respectively. Thus, $L = 6554/N$ bytes in the first case and $L = 16384/N$ bytes in the second case.

Figures 5-8 plot the M-UEP PSNR for all four groupings, versus N, for Lena image in each of the four channel conditions mentioned above. Each figure also contains the UEP PSNR and the M-UEP PSNR upper bound computed based on $\bar{D}_{opt}$. In all cases $R = 0.5$ bpp. The M-UEP PSNR upper bound always outperforms UEP at differences approaching 0.7 dB. Moreover, the performance of M-UEP using

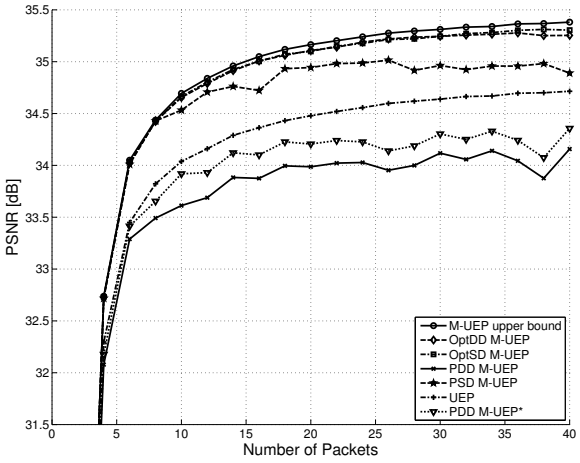

Fig. 7. **Grouping comparison:** M-UEP PSNR vs. number of packets, for at $R = 0.50$ bpp, for EPL $\epsilon = 0.05$.

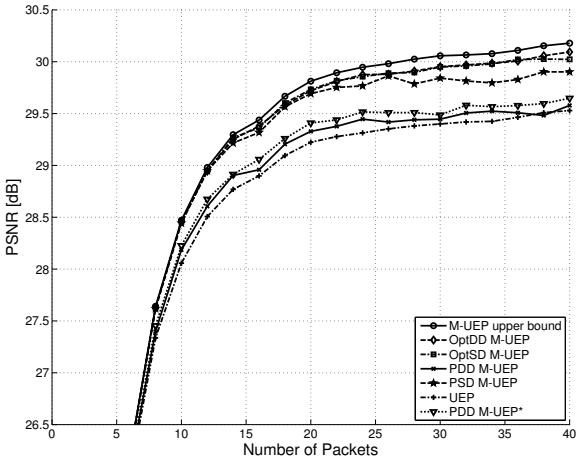

Fig. 8. **Grouping comparison:** M-UEP PSNR vs. number of packets, for at $R = 0.50$ bpp, for EPL $\epsilon = 0.15$.

optimized groupings is very close to the upper bound, suggesting that the grouping optimization criterion ensures good balance properties to the multi-stream. Interestingly, the performance of OptSD and OptDD-based M-UEP is almost identical.

The performance of PSD-based M-UEP is close to OptSD for smaller values of N, but degrades comparing to OptSD as N increases. This degradation is higher for IPL channel than for EPL channel, and for each channel it accentuates as the mean packet loss rate decreases. Notably, the PSD-based M-UEP performance is always above UEP level, except for the case of IPL channel at small loss rate and high N. Finally, PDD performs the worst and it is inferior to UEP in all cases except for EPL channel

with higher mean loss rate. The above observations remain consistent for Lena at $R = 0.2$ bpp and for Peppers at $R = 0.20$ and $R = 0.50$ bpp, but due to space limitations we do not include those plots here. The poor performance of PDD was expected since PDD generates some streams much shorter than others thus leading to severe imbalance. This is because in DD ordering, primary streams corresponding to non-root nodes (which are much shorter than the others) are consecutive. For severely unbalanced multi-streams the proposed M-UEP redundancy allocation algorithm does not offer performance guarantees. To verify this statement we also implemented the redundancy allocation algorithm which uses in Step 2 the layer sizes of the optimal UEP. The plots labeled PDD M-UEP* inserted in Figures 5-8 show the PSNR achieved with this rate allocation method, in the case od PDD grouping. In all cases, PDD M-UEP* outperforms PDD M-UEP, and the gap can reach up to 0.3 dB. However PDD M-UEP* still remains below UEP level in the case of IPL and EPL with small mean loss rate. Determining a close to optimal solution for severely unbalanced streams and clarifying if the M-UEP strategy maintains the benefit over UEP in such cases, remains a topic for further research.

To better highlight the gain of M-UEP over UEP we plot in Figures 9, 10, the difference in PSNR between OptSD M-UEP and UEP, versus the number of packets, for the IPL and EPL channels respectively. Each figure contains four plots corresponding to both images at each loss rate. $R = 0.5$ bpp in all cases. These results show that the gain of M-UEP over UEP can reach up to 0.6 dB. Interestingly, the gains are higher in the case of EPL than IPL, at the same loss rate, especially as N increases. In order to understand these phenomenon, it is useful to analyze the performance between M-UEP and UEP when only a fraction of the total number of packets is received. Figures 11, 12 plot the PSNR [dB] value achieved when only a subset of $k < N$ packets are received for OptSD M-UEP and UEP, for IPL and EPL respectively. Both figures illustrate the performance of M-UEP for Peppers, at $R = 0.5$ bpp, $N = 16$, and packet loss rate $\epsilon = \mu = 0.15$. We observe that, as a result of optimization of both UEP and M-UEP schemes, the advantage of M-UEP is higher when the number of lost packets is farther away from the the mean number of lost packets. Given a fixed loss rate $\epsilon = \mu$, the probability of this event is higher for EPL channels, thus leading to higher PSNR gains on average.

As discussed in Section V the performance advantage of M-UEP over UEP comes at the cost of additional side information needed at the decoder. In Figure 13 we plot the ratio between the amount of M-UEP side information and R versus N, for Lena for both channel models, at both loss rates. We also include the plot corresponding to the upper bound based on (12), and the plot corresponding to the UEP side information. The transmission budget is 0.5 bpp and OptSD grouping is considered. As it can be seen, the amount of side information is higher for the EPL channel comparing to IPL, and for

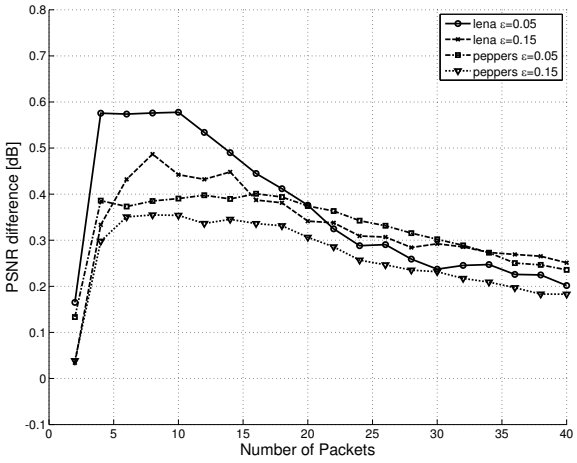

Fig. 9. **M-UEP vs. UEP:** PSNR difference between OptSD M-UEP and UEP vs. number of packets, for IPL at $R = 0.50$ bpp.

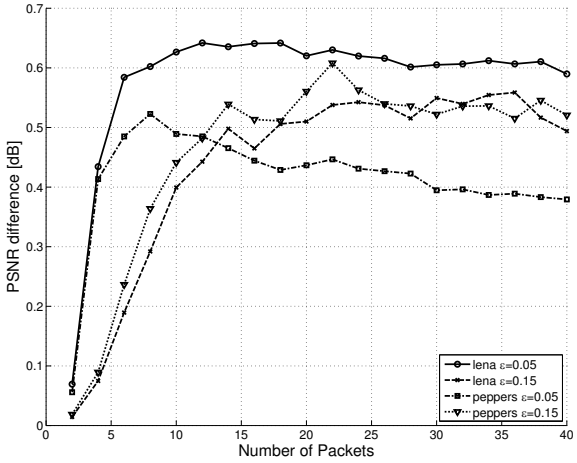

Fig. 10. **M-UEP vs. UEP:** PSNR difference between OptSD M-UEP and UEP vs. number of packets, for EPL at $R = 0.50$ bpp.

both channel models the overhead increases with the loss rate. The reason for this behaviour is that the M-UEP optimization allocates more non-zero layers in the case of EPL than IPL, at the same loss rate. Furthermore, for each channel model, more non-zero layers are allocated when the loss rate increases. We also observe that the ratio between the amount of M-UEP side information and R increases roughly linearly with N, but the rate of increase is lower than predicted by the upper bound (12).

The increase in the amount of M-UEP side information with N could render M-UEP unattractive at high number of packets. Therefore, techniques to decrease the overhead due to this side information are

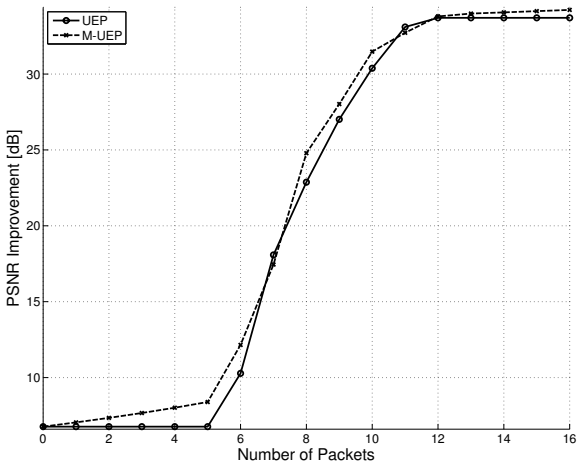

Fig. 11. **M-UEP vs. UEP:** PSNR of OptSD M-UEP and UEP vs. number k of received packets, for peppers, at $R = 0.50$ bpp, $N = 16$, IPL with $\epsilon = 0.15$.

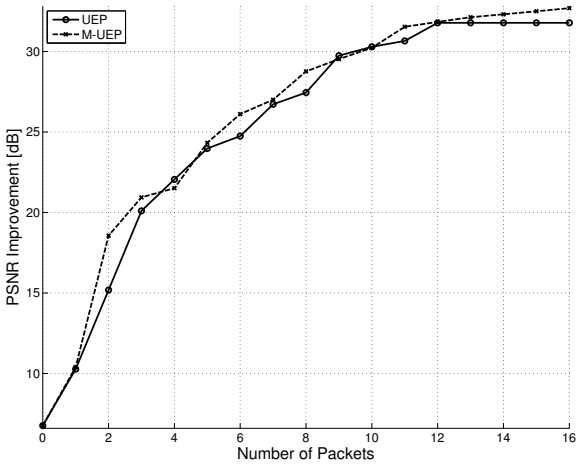

Fig. 12. **M-UEP vs. UEP:** PSNR of OptSD M-UEP and UEP vs. number k of received packets, for peppers, at $R = 0.50$ bpp, $N = 16$, EPL with $\mu = 0.15$.

of interest. One possibility is for each j, to force the values $x_j^{(i)}$ for all i to be equal. In this case it is enough to transmit to the decoder only the size of each layer x_j. The decoder derives the total amount of source symbols in layer j corresponding to each packet i as $x_j^{(i)} = jx_j/N$. If the division jx_j/N is not exact, then some $x_j^{(i)}$'s are set to $\lfloor jx_j/N \rfloor$ and the others to $\lceil jx_j/N \rceil$, according to a fixed rule. We refer to such a constrained M-UEP as M-UEP with fixed redundancy (FM-UEP, for short) because fixed redundancy is allocated across packets in each layer. The FM-UEP side information is the same as in UEP since it is enough to transmit to the decoder only the size of each layer x_j.

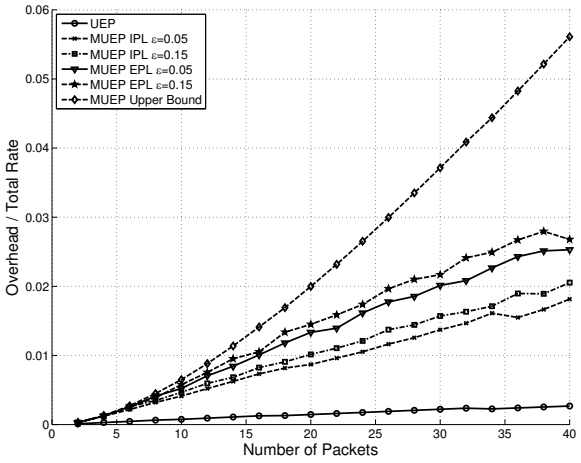

Fig. 13. **M-UEP side information:** Ratio between M-UEP side information length and total bit budget, vs. number of packets, for lena, at $R = 0.50$ bpp.

We have implemented the FM-UEP scheme in conjunction with the four groupings. Our tests reveal the following. First of all, OptSD grouping remains the best among all four groupings and is always superior to UEP for the IPL channel at both erasure rates and for the EPL channel at low mean loss rate. For these channel conditions, the drop in performance versus the M-UEP counterpart is modest, about 0.1 dB for IPL channel and 0.2 dB for the EPL channel. For the same channel conditions, OptDD-based FM-UEP is worse than OptSD in most of the cases, but the gap is small. Interestingly, both OptSD and OptDD are more dramatically affected by the fixed redundancy constraint in the case of EPL at high loss rate, when their PSNR decreases below UEP for all values of N. As for PSD grouping, it is more affected by the restriction of fixed redundancy in the case of IPL at high N. In this case its PSNR drops below UEP level even for high loss rate. On the other hand, PSD is the most robust among the four groupings to the restriction of fixed redundancy, in the case of EPL channel. For this type of channel, PSD FM-UEP performs better than UEP at both loss rates it, and most notably, at high mean loss rate, PSD is the only grouping which succeeds to maintain the performance of FM-UEP still higher than UEP. Surprisingly, in this case OptSD FM-UEP's PSNR becomes inferior to UEP as N increases, at a gap of up to 1dB. The above observations are consistent for both image, at both transmission rates. For illustration we plot in Figures 14 and 15 the PSNR values of UEP and FM-UEP with the four groupings, versus N for IPL, and EPL channel models respectively. In both figures image Lena at 0.5 bpp is considered and the channel loss rate is 0.15. To illustrate the gains of FM-UEP versus UEP we also plot in Figures 16, 17, the difference in PSNR between FM-UEP and UEP for the IPL and EPL

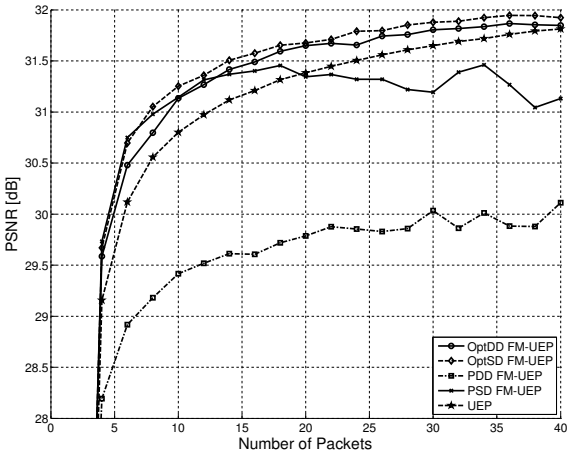

Fig. 14. **Grouping comparison:** PSNR of FM-UEP vs. number of packets, for lena, at $R = 0.50$ bpp, for IPL with $\epsilon = 0.15$.

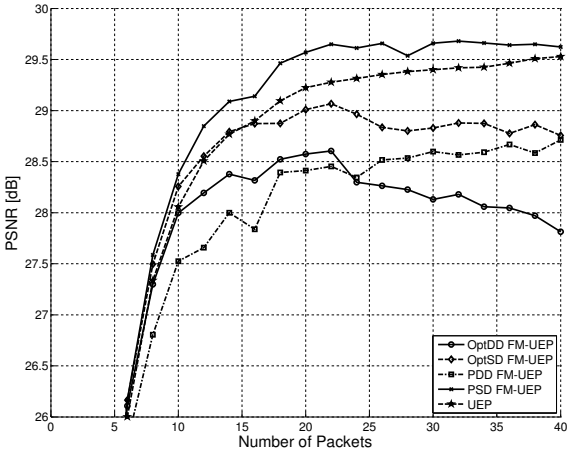

Fig. 15. **Grouping comparison:** PSNR of FM-UEP vs. number of packets for lena, at $R = 0.50$ bpp, for EPL with $\mu = 0.15$.

channel respectively. Each figure contains four plots corresponding to both images, for $R = 0.5$bpp at both loss rates. For each channel condition we consider FM-UEP in conjunction with the grouping which has the best performance, precisely, PSD for EPL with $\mu = 0.15$, and OptSD in all other cases.

It is also instructive to assess the performance of the M-UEP strategy with respect to UEP under channel mismatch conditions. Figure 18 plots the PSNR difference between OptSD M-UEP and UEP for each image and each channel model, under channel mismatch conditions. The actual channel loss rate varies in the range $[0.01 - 0.30]$, while the redundancy allocation for both transmission schemes is optimized for the loss rate $\epsilon = \mu = 0.15$. The number of packets is $N = 16$ and $R = 0.5$ bpp. It is clear from these figures that OptSD M-UEP maintains its superiority over UEP in all cases. Another

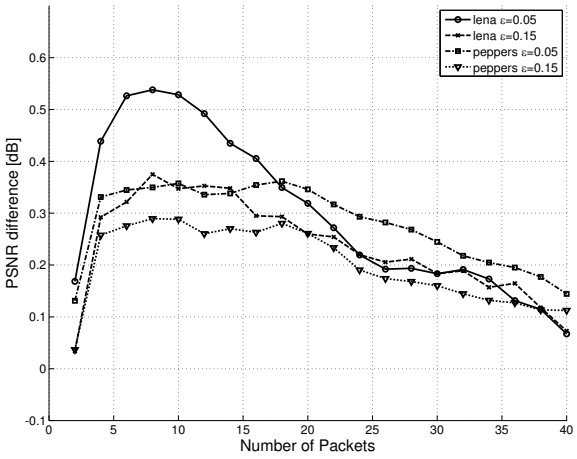

Fig. 16. **FM-UEP vs. UEP:** PSNR difference between FM-UEP and UEP vs. number of packets, for IPL at $R = 0.50$ bpp.

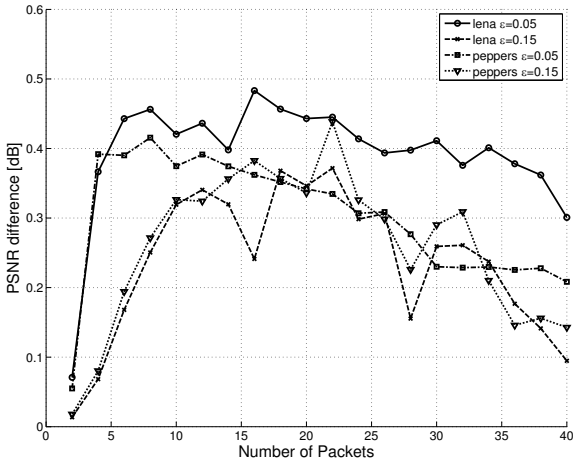

Fig. 17. **FM-UEP vs. UEP:** PSNR difference between FM-UEP and UEP vs. number of packets, for EPL at $R = 0.50$ bpp.

interesting observation is that for most images the improvement in PSNR increases as the erasure rate tends away from the predicted value for IPL, while for EPL increases only as the erasure rate becomes smaller than the predicted value. We believe that the increase in performance as the loss rate becomes smaller can be attributed to the fact that our redundancy allocation algorithm for M-UEP assigns less total redundancy than UEP. Consequently, M-UEP has higher performance than UEP when the number of received packets is very high. As the packet loss rate decreases, the probability of this event becomes higher leading to an increase in PSNR improvement.

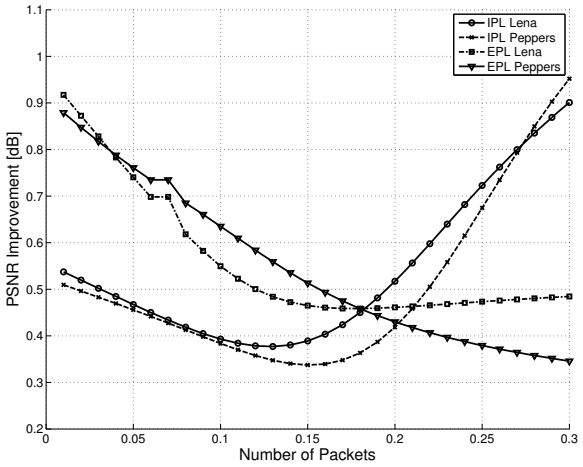

Fig. 18. **Channel mismatch performance :** PSNR improvement [dB] of BB M-UEP over UEP vs. packet erasure rate ϵ for all tested images at transmission rate $R = 0.50$ bpp and $N = 8$. The erasure protection is optimized for erasure rate $\epsilon = 0.15$.

VII. CONCLUSION

This paper proposed a new uneven erasure protection (UEP) strategy, termed M-UEP, for the transmission of multi-streams over packet erasure networks. M-UEP creates independently decodable packets to ensure that all received source symbols are decoded and uses permuted Reed-Solomon codes to increase the flexibility of the redundancy assignment. The R-D optimal redundancy allocation (RD ORA) problem was formulated and shown to have a time complexity of $O(2^N N(L+1)^{N+1})$, where N is the number of transmitted packets and L is the packet size. To address the high complexity of the globally optimal solution an efficient sub-optimal algorithm running in $O(N^2L^2)$ time was proposed. The additional side information necessary for M-UEP at the decoder was discussed and an upper bound on the side information length was derived. Moreover, a technique for mitigating the side information (FM-UEP) was presented. Experiments performed on SPIHT coded images (with appropriate grouping of wavelet coefficient) validated the superiority of M-UEP and FM-UEP over UEP, with peak improvements of $0.6, 0.5$ dB, respectively. Additionally, our tests revealed that M-UEP is more robust than UEP in adverse, unpredictable and varying channel conditions. Future research interests include improving the solution to M-UEP RD ORA problem and applying M-UEP to JPEG2000 coded images. Initial upper-bound experiments with JPEG2000 reveal the promise of similar performance improvement of M-UEP over UEP as achieved with the SPIHT coder.

APPENDIX A
SUFFICIENCY OF M-UEP RD-ORA CONSTRAINTS (4), (5), (6)

In this appendix we show that the constraints (4), (5), (6) are sufficient for the M-UEP RD-ORA problem formulated in Section 3. To this end we need to prove that for any non-negative integers $x_j, x_j^{(i)}, 1 \leq i, j \leq N$, satisfying conditions (4), (5), (6), there is a corresponding arrangement of the source symbols in the packetization array compatible with the M-UEP framework.

Proof. We begin by fixing some j and considering populating the j-th layer of the packetization array with source symbols. Consider N containers, where the i-th container has $x_j^{(i)}$ items. Populating the j-th M-UEP layer is equivalent to placing all these items in an array of x_j rows and N columns such that all items of the i-th container to be situated in the i-th column, and each row to contain exactly j items. The algorithm to perform this task proceeds in x_j iterations, each iteration populating one row. At the beginning of each iteration the containers are sorted in non-increasing order of the sizes of their contents. Then an item is withdrawn from each of the first j containers. Each item is placed in the current row in the column corresponding to the container.

To prove the algorithm correctness we need to show that at the beginning of each iteration there are at least j non-empty containers. If this result is established then it is clear that at algorithm completion all containers are empty, and hence all items have been placed in the array as required.

To validate the above claim we prove a stronger statement. Namely, we show that at the beginning of each iteration $\ell, 1 \leq \ell \leq x_j$, each container has at most $x_j - \ell + 1$ items and there are at least j non empty containers. For this we present a proof by induction over ℓ. When $\ell = 1$ the first part of the claim obviously holds by condition (6). Further, assuming that less than j containers are non empty, it follows that the total number of items is at most $(j-1)x_j$, which contradicts relation (5). Consequently, the second part of the claim is satisfied as well.

Now we proceed to the inductive step. Assume that the claim holds for some $\ell, 1 \leq \ell \leq x_j - 1$, and let us prove it for $\ell + 1$. By the inductive hypothesis, at the beginning of the ℓ-th iteration each container has at most $x_j - \ell + 1$ items. This further implies that at most j containers can have exactly $x_j - \ell + 1$ items. Indeed, by assuming the contrary we obtain the total number of items still in the containers to be larger or equal than $(j+1)(x_j - \ell + 1)$ which contradicts the fact that that exactly $j(\ell - 1)$ items have already been withdrawn. Consequently, all those containers having exactly $x_j - \ell + 1$ items (if any) will have an item withdrawn during the ℓ-th iteration. Thus, at the beginning of the $(\ell + 1)$-th iteration

no container has $x_j - \ell$ or more items. Thus, the first part of the claim is proven. Assuming now that less than j containers are non empty at the beginning of the $(\ell+1)$-th iteration, we obtain that the total number of items is at most $(j-1)(x_j - \ell)$ items, which contradicts the fact that that exactly $j\ell$ items have already been withdrawn. With this, the proof is completed. □

APPENDIX B
GRAPH-BASED FORMULATION OF M-UEP RD-ORA PROBLEM

In this appendix, we derive a weighted directed acyclic graph G and show that the M-UEP RD-ORA problem is equivalent to the maximum-weight path problem in G.

Let $G = (V, E)$, where V is the set of vertices (or nodes) and E is the set of directed edges. The nodes of the graph are all $(N+2)$-tuples of integers $(u^{(1)}, u^{(2)}, \ldots, u^{(N)}, n, \ell)$ such that $0 \leq \ell \leq L$, $0 \leq u^{(i)} \leq \ell$, for all $i \in \{1, 2, \ldots, N\}$, and $0 \leq n \leq N$. The edges of the graph are all ordered pairs of vertices $(\mathbf{u}, \mathbf{v})$, where $\mathbf{u} = \left(u^{(1)}, u^{(2)}, \ldots, u^{(N)}, n, \ell\right)$ and $\mathbf{v} = \left(v^{(1)}, v^{(2)}, \ldots, v^{(N)}, m, \kappa\right)$ such that the following conditions are satisfied

$$\kappa = \ell + 1, \quad m \geq n \tag{13}$$

$$0 \leq v^{(i)} - u^{(i)} \leq 1 \,\forall\, i, 1 \leq i \leq N, \tag{14}$$

$$\textstyle\sum_{i=1}^{N} \left(v^{(i)} - u^{(i)}\right) = m. \tag{15}$$

Each edge $(\mathbf{u}, \mathbf{v})$ is assigned a weight $w(\mathbf{u}, \mathbf{v})$ defined as

$$w(\mathbf{u}, \mathbf{v}) = \sum_{i=1}^{N} \left(C_M(i, m)\,(v^{(i)} - u^{(i)}) \Delta D_i \left(v^{(i)} \right) \right). \tag{16}$$

The source node of the graph is $\mathbf{v}_0 = (0, 0, \ldots, 0, 0)$, and the final nodes are all vertices whose last component is L. A path in G is any sequence of nodes starting with the source node and ending with a final node, such that any two consecutive nodes are connected by an edge. The weight of the path is defined as the sum of the weights of its edges. Note that the last component of the i-th node of the path must necessarily be $i-1$. Therefore, any path in G has exactly $L+1$ nodes, and hence L edges.

Any M-UEP packetization array PA can be assigned a path P_{PA} in G as follows. For each row ℓ $(1 \leq \ell \leq L)$, and each column i $(1 \leq i \leq N)$, let $v_\ell^{(i)}$ denote the number of source symbols in the first ℓ rows of column i, and let n_ℓ denote the layer corresponding to that row. Let $\mathbf{v}_\ell = \left(v_\ell^{(1)}, v_\ell^{(2)}, \ldots, v_\ell^{(N)}, n_\ell, \ell\right)$, $1 \leq \ell \leq L$, be the vertex representing the source symbol assignment in the first ℓ rows of the packetization array. Then the path P_{PA} is defined as the sequence of vertices $\mathbf{v}_0, \mathbf{v}_1, \ldots, \mathbf{v}_L$ (it is easy to see that any ordered pair $(\mathbf{v}_{\ell-1}, \mathbf{v}_\ell)$ forms an edge). Note that the ℓ-th path edge $(\mathbf{v}_{\ell-1}, \mathbf{v}_\ell)$ corresponds to the ℓ-th

row in the packetization array. Moreover, the weight of this edge equals the contribution of the source symbols in the ℓ-th row to the decrease of $\bar{D}_M$ in (8). Therefore, the weight of the path $w(P_{PA})$ satisfies

$$w(P_{PA}) = D_{max} - \bar{D}_M. \tag{17}$$

It is easy to see that the above correspondence between M-UEP packetization arrays and paths in the graph G is one-to-one. Therefore, we conclude that the M-UEP RD-ORA problem is equivalent to the maximum-weight path problem in the graph G.

The time complexity to solve the latter problem is $O(|V|+|E|)$. The number of vertices $|V|$ is clearly $O(N(L+1)^{N+1})$. In order to determine $|E|$ let us first evaluate the number of edges $(\mathbf{u}, \mathbf{v})$ outgoing from some node $\mathbf{u}$. Note that the last component of $\mathbf{v}$ is determined by $\mathbf{u}$. For any possible value m of the second last component, the number of choices of $v^{(1)}, v^{(2)}, \ldots, v^{(N)}$, which satisfy (15) is $\binom{N}{m}$. It follows that the total number of nodes $\mathbf{v}$ is $\sum_{m=n}^{N} \binom{N}{m} \leq 2^N$. Consequently, $|E| = O(|V| \times 2^N)$. We conclude that the globally optimal solution to the problem of R-D optimal M-UEP packetization can be found in $O(2^N N(L+1)^{N+1})$ time.

APPENDIX C
SYMBOL DECODING PROBABILITY FOR SYMMETRIC CHANNELS

Denote $\mathcal{N} = \{1, 2 \cdots, N\}$. For any $\mathcal{I} \subseteq \mathcal{N}$, let $Q_N(\mathcal{I})$ denote the probability that the packets in subset $\mathcal{I}$ are received, while the packets in $\mathcal{N} - \mathcal{I}$ are lost. Assume a symmetric packet loss channel as defined in Section IV. Since $Q_N(\mathcal{I}) = Q_N(\mathcal{I}')$ for any $\mathcal{I}, \mathcal{I} \subseteq \mathcal{N}$, with equal number of elements, it follows that

$$Q_N(\mathcal{I}) = \frac{P_N(N-s)}{\binom{N}{s}}, \tag{18}$$

where s is the size of $\mathcal{I}$. Let μ denote the average number of lost packets (out of N transmitted packets). Recall that $\alpha(i)$ denotes the probability that packet i is received. Then we obtain

$$\alpha(i) = \textstyle\sum_{\mathcal{I} \subseteq \mathcal{N}, i \in \mathcal{I}} Q_N(\mathcal{I}) =$$

$$\textstyle\sum_{k=0}^{N-1} \sum_{\mathcal{I} \subseteq \mathcal{N}, |\mathcal{I}| = N-k, i \in \mathcal{I}} Q_N(\mathcal{I})$$

$$= \textstyle\sum_{k=0}^{N-1} \frac{P_N(k)}{\binom{N}{N-k}} \binom{N-1}{N-k-1} =$$

$$\textstyle\sum_{k=0}^{N-1} P_N(k) \frac{N-k}{N} = \sum_{k=0}^{N} P_N(k)(1 - k/N) = 1 - \mu.$$

The third equality in the above sequence follows from (18). Further, recall that $\beta(i,j)$ denotes the probability that packet i is lost and at least j other packets are received. Then we have

$$\begin{aligned}\beta(i,j) &= \sum_{k=0}^{N-j} \sum_{\mathcal{I}\subseteq\mathcal{N},|\mathcal{I}|=N-k,i\notin\mathcal{I}} Q_N(\mathcal{I}) \\ &= \sum_{k=0}^{N-j} \frac{P_N(k)}{\binom{N}{N-k}} \binom{N-1}{N-k} = \sum_{k=0}^{N-j} \frac{k}{N} P_N(k),\end{aligned}$$

where the second equality follows from (18).

The above relations imply that $\alpha(i)$ and $\beta(i,j)$ are constant in i.

REFERENCES

[1] A. Said and W. Pearlman, "A new fast, and efficient image codec based on set partitioning in hierarchical trees," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, pp. 243–250, Jun. 1996.

[2] D. Taubman, "High performance scalable image compression with EBCOT," *IEEE Trans. Image Process.*, vol. 9, pp. 1158–1170, Jul. 2000.

[3] A. Mohr, E. Riskin, and R. Ladner, "Graceful degradation over packet erasure channels through forward error correction," in *Proc. DCC'99*, Utah, USA, Mar. 1999, pp. 92–101.

[4] R. Puri and K. Ramchandran, "Multiple description source coding through forward error correction codes," in *Proc. 33rd Asilomar Conference on Signals, Systems, and Computers*, vol. 1, California, USA, Oct. 1999, pp. 342–346.

[5] A. E. Mohr, E. A. Riskin, and R. E. Ladner, "Unequal Loss Protection: Graceful Degradation over Packet Erasure Channels through Forward Error Correction", *IEEE Journal on Selected Areas in Communication*, vol. 18, no. 7, pp. 819–828, Jun. 2000.

[6] T. Stockhammer, C. Buchner, "Progressive texture video streaming for lossy packet networks", *Proc. 11th International Packet Video Workshop*, Kyongju, May 2001.

[7] V. Stankovic, R. Hamzaoui, and Z. Xiong, "Efficient channel code rate selection algorithms for forward error correction of packetized multimedia bitstreams in varying channels," *IEEE Trans. Multimedia*, vol. 14, no. 2, pp. 240–248, Apr. 2004.

[8] S. Dumitrescu, X. Wu, and Z. Wang, "Globally optimal uneven error-protected packetization of scalable code streams," *IEEE Trans. Multimedia*, vol. 6, no. 2, pp. 230–239, Apr. 2004.

[9] J. Thie and D. Taubman, "Optimal erasure protection strategy for scalably compressed data with tree-structured dependencies," *IEEE Trans. Image Process.*, vol. 14, no. 12, pp. 2002–2011, Dec. 2005.

[10] S. Dumitrescu, X. Wu, and Z. Wang, "Efficient Algorithms for Optimal Uneven Protection of Single and Multiple Scalable Code Streams against Packet Erasures," *IEEE Trans. Multimedia*, vol. 9, no. 7, pp. 1466-1474, Nov. 2007.

[11] R. Hamzaoui, V. Stankovic, and Z. Xiong, "Optimized error protection of scalable image bit streams : Advances in joint source-channel coding of images," *IEEE Signal Processing Magazine* , no. 6, pp. 91–107, Nov. 2005.

[12] C. D. Creusere, "A new method of robust image compression based on the embedded zerotree wavelet algorithm," *IEEE Trans. Image Process.*, vol. 6, no. 10, pp. 1436–1442, Oct. 1997.

[13] J. Shapiro, "Embedded image coding using zerotrees of wavelet coefficients," *IEEE Trans. Signal Process.*, vol. 41, pp. 3445–3462, Dec. 1993.

[14] S. Cho and W. Pearlman, "Error resilient compression and transmission of scalable video," in *Proc. SPIE*, vol. 4115, San Diego, USA, Jul. 2000, pp. 396–405.

[15] A. A. Alatan, M. Zhao, and A. N. Akansu, "Unequal error protection of SPIHT encoded image bit streams," *IEEE J. Sel. Areas Commun.*, vol. 18, pp. 814–818, Jun. 2000.

[16] J. Kim, R. M. Mersereau, and Y. Altunbasak, "Error-resilient image and video transmission over the Internet using unequal error protection," *IEEE Trans. Image Process.*, vol. 12, no. 2, pp. 121–131, Feb. 2003.

[17] J. Rogers and P. Cosman, "Robust wavelet zerotree image compression with fixed-length packetization," in *Proc. DCC'98*, Utah, USA, Mar. 1998, pp. 418–427.

[18] X. Wu, S. Cheng, and Z. Xiong, "On Packetization of Embedded Multimedia Bitstreams", *IEEE Trans. Multimedia*, vol. 3, no. 1, pp. 132-140, Mar. 2001.

[19] N. Thomos, N. Boulgouris, and M. Strintzis, "Optimized transmission of JPEG2000 streams over wireless channels," *IEEE Trans. Image Process.*, vol. 15, no. 1, pp. 54–67, Jan. 2006.

[20] ——, "Product code optimization for determinate state LDPC decoding in robust image transmission," *IEEE Trans. Image Process.*, vol. 15, no. 8, pp. 2113–2119, Aug. 2006.