Globally Optimal Uneven Error-Protected Packetization of Scalable Code Streams

Sorina Dumitrescu, Xiaolin Wu Senior Member, IEEE, and Zhe Wang
Dept. of Electrical and Computer Engineering
McMaster University
Hamilton, ON, Canada L8S 4K1
sorina/xwu@mail.ece.mcmaster.ca, zwang@grads.ece.mcmaster.ca

Abstract. In this paper we present a family of new algorithms for rate-fidelity optimal packetization of scalable source bit streams with uneven error protection. In the most general setting where no assumption is made on the probability function of packet loss or on the rate-fidelity function of the scalable code stream, one of our algorithms can find the globally optimal solution to the problem in $O(N^2L^2)$ time, compared to a previously obtained $O(N^3L^2)$ complexity, where N is the number of packets and L is the packet payload size. If the rate-fidelity function of the input is convex, the time complexity can be reduced to $O(NL^2)$ for a class of erasure channels, including channels for which the probability function of losing n packets is monotonically decreasing in n and independent erasure channels with packet erasure rate no larger than $\frac{N}{2(N+1)}$. Furthermore, our $O(NL^2)$ algorithm for the convex case can be modified to find an approximation solution for the general case. All of our algorithms do away with the expediency of fractional bit allocation, a limitation of some existing algorithms.

Key words: Uneven error protection, multimedia streaming, joint source-channel coding, optimization.

EDICS: 5-STRM; Special Issue on Streaming Media.

1 Introduction

Modern packet switched communication systems such as ATM and the Internet have to overcome the problems of packet loss and other transmission errors. In the case of streaming a scalable source sequence of compressed digital media, one can use optimal packetization of the scalable source sequence with uneven error protection (UEP) to minimize the impact of lost packets on the quality of network service. The idea is to partition a scalable source sequence into segments of decreasing importance, and protect these segments by progressively weaker error correction codes to achieve the best joint economy of source and channel codes. This joint source-channel coding approach was recently studied by many researchers [2, 4, 5, 7, 8, 9, 10, 12, 13], most of whom were motivated by the applications of internet media streaming over noisy channels. The technique can be applied to any scalable (progressively refinable) signal compression methods. Many experiments with scalable image coding techniques such as SPIHT and JPEG 2000 were reported in the literature [4, 7, 8, 10, 12, 13, 14].

In this paper, we reexamine the problem of UEP optimal packetization of scalable source sequence of Reed-Solomon (RS) block codes as formulated and discussed in [5, 7, 8, 9, 14, 16], and develop new algorithms for it. In particular, we focus on globally optimal solutions to the UEP packetization problem in more general setting.

Consider transmission of a scalable source sequence using N packets, each of which has a payload of L symbols (a symbol is a block of a fixed number of bits). In the UEP framework, the source sequence is divided into L consecutive segments, and each of these segments is protected by RS code. Let m_i be the length (in symbols) of the i-th source segment, then the channel code assigned to protect the segment is the (N, m_i) RS code. The stream of these m_i source symbols followed by the $f_i = N - m_i$ redundancy symbols constitutes the i-th slice of the joint source-channel code. The effect of the (N, m_i) RS code associated with the i-th source segment is that, if at most f_i of N packets are lost, then all the m_i source symbols of the i-th slice can be correctly recovered. However, since the scalable source sequence is only sequentially refinable, decoding of the i-th source segment depends on all the previous $i - 1$ segments, i.e., the complete prefix of the source sequence with respect to the current segment. Hence the number of redundancy symbols assigned to a slice must

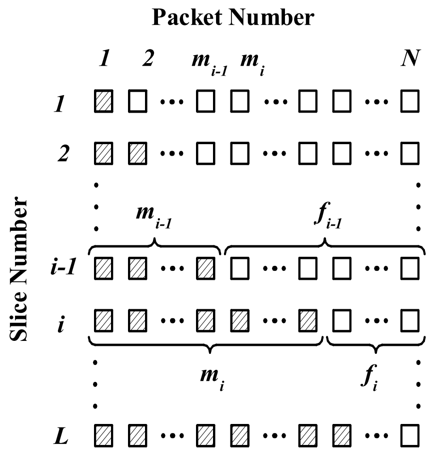

Figure 1: UEP packetization scheme. The dark rectangles correspond to source symbols and the white rectangles correspond to the redundancy symbols.

be monotonically non-increasing in the slice index:

$$f_1 \geq f_2 \geq \cdots \geq f_L. \tag{1}$$

The L-tuple $(f_1, f_2, \cdots, f_L)$ above is called L-slice redundancy assignment.

Since no fractional protection symbols can be allocated in practice, we require that all $f_i, 1 \leq i \leq L$ be integers between 0 and $N-1$ and that the monotone relation (1) be enforced. Figure 1 illustrates the UEP packetization scheme. As we will see later, the constraint of decreasing redundancy level for subsequent segments of the source sequence can be lifted, if the rate-fidelity function of the scalable source sequence is convex. In this case, the solution of the unconstrained version of the optimization problem satisfies the constraint anyway. In practice many scalable source codes are indeed constructed to shape an approximately convex operational rate-fidelity curve. A well-known example is the EBCOT technique used in JPEG 2000 [17].

Let $\phi(r)$ be the rate-fidelity function of the scalable source sequence, which is a monotonically non-decreasing function in rate $r \in [0, R_{max}]$, where R_{max} is the total number of source symbols. The efficiency of the redundancy assignment is measured by the expected fidelity of the reconstructed sequence at the decoder side. If exactly n packets are lost, such that $f_i \geq n > f_{i+1}$, for some i, then only the first i source segments can be completely recovered by the RS code. Since the source sequence is embedded, the receiver can decode it only up to the first lost symbol. Consequently, the receiver can decode only the first i source segments plus some few source symbols of the $(i+1)$-th segment that are not lost. As in [7, 8, 9, 14, 16], we round off the effect of decoding these additional few symbols on fidelity. Hence the achieved fidelity is $\phi(r_i)$, where $r_i = \sum_{k=1}^{i} m_k = iN - \sum_{k=1}^{i} f_k$. The probability that the receiver achieves this fidelity is $\sum_{n=f_{i+1}+1}^{f_i} p_N(n)$, where $p_N(n)$ is the probability of losing n packets out of N. Hence, the expected fidelity $\Phi(f_1, f_2, \cdots, f_L)$ of the reconstructed sequence at the decoder side can be expressed as

$$\Phi(f_1, f_2, \cdots, f_L) = \phi(0) \sum_{n=f_1+1}^{N} p_N(n) + \sum_{i=1}^{L-1} \phi(r_i) \sum_{n=f_{i+1}+1}^{f_i} p_N(n) + \phi(r_L) \sum_{n=0}^{f_L} p_N(n). \tag{2}$$

After straightforward algebraic manipulations we have:

$$\Phi(f_1, f_2, \cdots, f_L) = c_N(N)\phi(0) + \sum_{i=1}^{L} c_N(f_i)(\phi(r_i) - \phi(r_{i-1})), \tag{3}$$

where $c_N(k) = \sum_{n=0}^{k} p_N(n)$, $k = 0, 1, \cdots, N$, and $r_i = \sum_{k=1}^{i} m_k = iN - \sum_{k=1}^{i} f_k$, $1 \le i \le L$, $r_0 = 0$.

The objective of optimal UEP packetization is to find the redundancy assignment $(f_1, f_2, \cdots, f_L)$ that maximizes $\Phi(f_1, f_2, \cdots, f_L)$, for given N, L, $p_N(n)$, and $\phi(r)$.

Most of the existing algorithms proposed for this problem [7, 8, 9, 14, 16] need the convexity of the rate-fidelity function to achieve optimality. Stockhammer and Buchner [16] present a dynamic programming algorithm of $O(N^2L^2)$ time complexity that can obtain global optimality for convex rate-fidelity function. The algorithms of Puri and Ramchandran [9] and of Mohr, Ladner and Riskin [7, 8] provide the globally optimal solution only if the rate-fidelity function is convex and fractional bit allocation is allowed. The algorithm of Stankovic, Hamzaoui, and Xiong [14] does not need the additional assumption of fractional bit allocation, but it can find only a local optimum. The only known globally optimal solution of UEP packetization problem in general setting, with no assumptions on the rate-fidelity function or on the channel statistics, was given by Sachs, Anand and Ramchandran [10]. It was a dynamic programming algorithm of $O(N^3L^2)$ time complexity.

In this paper we reduce the time complexity of the globally optimal UEP packetization to $O(N^2L^2)$ for the general case, and to $O(NL^2)$ if the rate-fidelity function of the scalable source is convex and if the probability $p_N(n)$ is monotonically non-increasing in n. The condition of monotone $p_N(n)$ can be removed for independent packet erasure channels with packet loss rate no larger than $\frac{N}{2(N+1)}$. The paper is structured along this path of complexity reduction from the more general setting to the convex case. In the next section we propose an $O(N^2L^2)$ algorithm for globally optimal UEP packetization for the general case. In Section 3 we proceed to the case of convex rate-fidelity function. First we show that the constraint of non-increasing redundancy assignment (1) can be lifted. Then we prove that the cost function underlying the optimal unconstrained UEP packetization problem has a strong monotone property called total monotonicity if the probability $p_N(n)$ is monotonically non-increasing in n. This property allows the use of a fast matrix search technique [1] to reduce the time complexity to $O(NL^2)$. Section 4 briefly demonstrates how our algorithms can also be applied to optimize the product joint source-channel code as proposed by [10], which takes in consideration both packet loss and intra-packet bit errors. Section 5 presents experimental results and offers comprehensive comparisons between the proposed algorithms and existing approximation algorithms.

2 Exact Solution for the General Case

In this section we solve the optimal UEP packetization problem for general $\phi(r)$ and $p_N(n)$. Note that in (3) the term $c_N(N)\phi(0)$ is a constant and hence can be discarded in the optimal UEP design. In the summation that remains, the i-th term, $c_N(f_i)(\phi(r_i) - \phi(r_{i-1}))$, is the contribution of the i-th slice to the expected fidelity. The contribution of the first k slices to the expected fidelity, which depends only on the k-slice redundancy assignment $(f_1, f_2, \cdots, f_k)$, is denoted by $\Phi_k(f_1, f_2, \cdots, f_k)$:

$$\Phi_k(f_1, f_2, \cdots, f_k) = \sum_{i=1}^{k} c_N(f_i)(\phi(r_i) - \phi(r_{i-1})). \tag{4}$$

It is obvious that maximizing the expected fidelity $\Phi(f_1, f_2, \cdots, f_L)$ is equivalent to maximizing $\Phi_L(f_1, f_2, \cdots, f_L)$, which can be written as

$$\Phi_L(f_1, f_2, \cdots, f_L) = \Phi_k(f_1, f_2, \cdots, f_k) + \mathcal{C}, \tag{5}$$

where $\mathcal{C} = \sum_{i=k+1}^{L} c_N(f_i)(\phi(r_i) - \phi(r_{i-1}))$.

The algorithm development would be considerably simpler if the two terms $\Phi_k(f_1, f_2, \cdots, f_k)$ and $\mathcal{C}$ could be maximized separately. Unfortunately, the two expressions are not independent. $\mathcal{C}$ depends on the total number of protection symbols $f_1 + f_2 + \cdots + f_k$ for the first k slices (because $r_k = kN - (f_1 + f_2 + \cdots + f_k)$). Another dependency is imposed by the condition $f_k \geq f_{k+1}$ that has to be satisfied. We can break down these dependencies by fixing the total number of protection symbols t for the first k slices, $t = f_1 + f_2 + \cdots + f_k$, and by also fixing a value n satisfying $f_k \geq n \geq f_{k+1}$ (the value n has the significance of minimum redundancy required for the k-th slice and maximum redundancy admissible for the $(k+1)$-th slice). After introducing the new parameters n and t, consider the sub-problem of maximizing $\Phi_k(f_1, f_2, \cdots, f_k)$ over all k-slice redundancy assignments $(f_1, f_2, \cdots, f_k)$ for the first k slices, subject to $N-1 \geq f_1 \geq \cdots \geq f_k \geq n$, $f_1 + f_2 + \cdots + f_k = t$. We call this sub-problem, the sub-allocation (k, n, t). The sub-allocation (k, n, t) is defined for all triples (k, n, t) of integers such that $1 \leq k \leq L$, $0 \leq n \leq N-1$ and

$$kn \leq t \leq k(N-1). \tag{6}$$

For convenience of denotation, define

$$A(k,n,t) = \max \left\{ \Phi_k(f_1, f_2, \cdots, f_k) | N-1 \geq f_1 \geq \cdots \geq f_k \geq n, f_1 + f_2 + \cdots + f_k = t \right\}. \tag{7}$$

In order for such redundancy assignments to exist, the condition (6) has been imposed on the triple (k, n, t).

In the newly introduced notation the optimal UEP packetization problem can be restated as finding the L-slice redundancy assignment $(f_1, f_2, \cdots, f_L)$ such that

$$\Phi_L(f_1, f_2, \cdots, f_L) = \max_{0 \leq t \leq L(N-1)} A(L, 0, t). \tag{8}$$

In order to obtain the values $A(L, 0, t)$ for all possible t, we need to systematically evaluate $A(k, n, t)$ for all possible triples (k, n, t). The required computations are organized recursively as follows.

If the k-slice redundancy assignment $(f_1, f_2, \cdots, f_k)$ is optimal for the sub-allocation (k, n, t), then either $f_k = n$ or $f_k \geq n+1$. If $f_k = n$ then the total number of protection symbols on the first $k-1$ slices equals $t-n$ and the $(k-1)$-slice redundancy assignment $(f_1, f_2, \cdots, f_{k-1})$ has to be optimal for the sub-allocation $(k-1, n, t-n)$, too. If $f_k \geq n+1$, then the assignment $(f_1, f_2, \cdots, f_k)$ has to be optimal for the sub-allocation $(k, n+1, t)$ as well.

Note further that, for some triples (k, n, t) one of the two alternatives mentioned above ($f_k = n$ or $f_k \geq n+1$) is impossible, hence only the other holds. This is the case when $kn \leq t < k(n+1)$, which implies that f_k can not be larger than n, hence we obtain

$$A(k,n,t) = A(k-1, n, t-n) + c_N(n)(\phi(kN-t) - \phi((k-1)N - t + n)). \tag{9}$$

The other case happens when $n + (k-1)(N-1) < t \le k(N-1)$, which implies that f_k can not be equal to n, and it follows that

$$A(k, n, t) = A(k, n+1, t). \tag{10}$$

In all the other cases, i.e. for $k(n+1) \le t \le n + (k-1)(N-1)$, the following recursion holds:

$$A(k, n, t) = \max\{A(k-1, n, t-n) + \\ c_N(n)(\phi(kN-t) - \phi((k-1)N - t + n)), A(k, n+1, t)\}. \tag{11}$$

The recursive formulae (9), (10) and (11) show that $A(k, n, t)$ can be computed in constant time provided that $A(k-1, n, t-n)$ and $A(k, n+1, t)$ are known. In order to take advantage of this result we have to solve the sub-allocations (k, n, t) in such an order as to ensure that sub-allocation (k, n, t) is computed after sub-allocations $(k-1, n, t-n)$ and $(k, n+1, t)$. This can be done if k is enumerated in increasing order, but n in decreasing order, and for each given pair k, n, all possible t are considered before going to the next value of k or n.

After computing $\max_{0 \le t \le L(N-1)} A(L, 0, t)$, we need to restore the optimal L-slice redundancy assignment. Let $B(k, n, t)$ be the number of protection symbols for the k-th slice in the optimal k-slice redundancy assignment achieving $A(k, n, t)$ in (7), for each triple (k, n, t). We need to keep track of the values $B(k, n, t)$. To save space we do not store the actual values of $B(k, n, t)$ for each triple (k, n, t), but only a binary flag on which value is the maximum in the righthand side of (11). Thus, we define $Z(k, n, t)$ to be 0 if (9) holds, and 1 otherwise, for all $1 \le k \le L$, $N-1 \ge n \ge 0$, and $kn \le t \le k(N-1)$. Upon computation of $A(k, n, t)$, the bit $Z(k, n, t)$ is also set and stored. The binary array $Z(\cdot, \cdot, \cdot)$ suffices to reconstruct any quantity $B(k, n, t)$. Namely, $B(k, n, t) = n_1$, where n_1 is the smallest integer satisfying the conditions: $n \le n_1 \le t/k$, $Z(k, n_1, t) = 0$ and $Z(k, n', t) = 1$ for all $n', n \le n' < n_1$. Such an integer n_1 always exists (indeed, for $n" = \lfloor t/k \rfloor$ we have $Z(k, n", t) = 0$). Hence, for a given triple (k, n, t), the reconstruction of $B(k, n, t)$ requires at most $O(N)$ time.

Summarizing the above we present an UEP packetization algorithm to compute globally optimal L-slice redundancy assignment for arbitrary $p_N(n)$ and $\phi(r)$.

Algorithm A. Optimal UEP Packetization for General Case.

Step 1. For $k = 1, 2, \cdots, L$, $n = N-1, N-2, \cdots, 0$, and each t, $kn \le t \le k(N-1)$, compute $A(k,n,t)$ and $Z(k,n,t)$ using recursion (9), (10) or (11), depending on t.

Step 2. Find $t_0 = \max^{-1}_{0 \le t \le L(N-1)} A(L, 0, t)$.

Step 3. Construct the optimal assignment $(f_1, f_2, \cdots, f_L)$ in the following way. Set $f_L = B(L, 0, t_0)$ and $t = t_0 - f_L$. For $k = L-1, L-2, \cdots, 1$, set $f_k = B(k, f_{k+1}, t)$ and then update t: $t = t - f_k$. The values $B(k,n,t)$ are determined as explained earlier.

In Appendix A we present a detailed pseudocode of the algorithm.

The time complexity of Step 1 is $O(N^2L^2)$ because there are $O(N^2L^2)$ entries of $A(k,n,t)$ and $Z(k,n,t)$, each of which is computed in $O(1)$ time. Step 2 clearly takes $O(NL)$ operations. Step 3 restores L quantities $B(k,n,t)$ each of which requires $O(N)$ time, taking $O(NL)$ time in all. Consequently, the overall time complexity of the algorithm is $O(N^2L^2)$. The previously known algorithm for the same problem has a time complexity of $O(N^3L^2)$ [10].

Next we analyze the space complexity. For each triple (k,n,t), $Z(k,n,t)$ has to be stored until the algorithm is completed. Since there are $O(N^2L^2)$ such binary entries, $O(N^2L^2)$ bits suffice to store $Z(\cdot,\cdot,\cdot)$. The values of $A(k,n,t)$ are not needed in the entire duration of Step 1, but only as long as they are used in (9), (10) or (11). Among the iterations on k, n and t, k is the last one to vary. Therefore, after all quantities $A(k_0,n,t)$, for a fixed k_0, have been computed, the values $A(k,n,t)$ for $k < k_0$ and any n and t, are no longer needed and can be discarded. This means that only $O(N^2L)$ entries of $A(k,n,t)$ need to be stored at any given time. Consequently, the space complexity of the algorithm is dominated by the memory requirement of $Z(\cdot,\cdot,\cdot)$, which is $O(N^2L^2)$ in bits.

3 Fast Matrix-search Algorithm for Convex Case

The complexity of Algorithm A is still high. There are two ways to simplify Algorithm A and reduce its complexity. The first is to remove the constraint of non-increasing redundancy assignment to subsequent slices of the scalable source sequence in the optimization process, and hopefully without compromising optimality by doing so (this approach was taken by Stockhammer and Buchner [16], too). The second is to reduce the search space of the dynamic programming algorithm by the technique of fast matrix search [1]. This can be made possible if the underlying cost function satisfies a so-called total monotonicity. In this section we take these two steps to develop a more efficient algorithm for optimal UEP packetization.

The algorithm to be presented below finds the globally optimum of the UEP packetization problem in the case of convex rate-fidelity function, in a similar approach as the algorithm of [16] with $O(N^2L^2)$ time complexity. But we go a step further and show that its time complexity can be reduced to $O(NL^2)$ under some mild assumptions about the channel statistics, namely if $p_N(n)$ is monotonically decreasing in n or if the channel is an independent erasure channel with packet loss rate ϵ no greater than $\frac{N}{2(N+1)}$.

The solution presented in the previous section for optimal UEP packetization problem is found by recursively solving the sub-problems associated with parameters k, n, t. The parameter n was introduced to enforce the decreasing redundancy assignment, i.e., $f_1 \geq f_2 \geq \cdots \geq f_L$. If this condition is removed, then the parameter n can be dropped. In order to solve the problem of maximizing (3) without the constraint of (1), we recursively solve sub-problems associated with pairs of integers (k, t) with $1 \leq k \leq L, 0 \leq t \leq k(N-1)$. For each such pair (k, t) denote by $C(k, t)$ the analog of $A(k, 0, t)$ without the constraint (1). Namely,

$$C(k,t) = \max\{\Phi_k(f_1, \cdots, f_k) \mid 0 \leq f_1, \cdots, f_k \leq N-1, f_1 + \cdots + f_k = t\} \tag{12}$$

It is clear that $C(1, t) = \Phi_1(t)$ for all $t, 0 \leq t \leq N-1$.

The computation of $C(k, t)$ can be done according to the recursive formula

$$C(k,t) = \max_{0 \leq t-j \leq N-1} (C(k-1, j) + c_N(t-j)(\phi(kN-t) - \phi((k-1)N-j))) \tag{13}$$

for all $2 \leq k \leq L, 0 \leq t \leq k(N-1)$.

Denote by $\jmath(k, t)$ the largest value of j for which $C(k, t)$ is achieved in (13), for $2 \leq k \leq L, 0 \leq t \leq k(N-1)$, and set by convention $\jmath(1, t) = 0$ for $0 \leq t \leq N-1$. If there are more than one k-slice redundancy assignments $(f_1, \cdots, f_k)$ achieving $C(k, t)$ in (12), then we choose the one for which $f_1 + \cdots + f_{k-1} = \jmath(k, t)$, hence for which the number of redundancy symbols on the k-th slice, f_k, is the smallest.

Consequently, the unconstrained version of the optimal UEP packetization problem can be solved by recursively computing the values $C(k, t)$ and $\jmath(k, t)$ for all $1 \leq k \leq L$ and $0 \leq t \leq k(N-1)$.

Then $t_0 = \max^{-1}_{0 \le t \le L(N-1)} C(L,t)$ is found and the optimal L-slice redundancy assignment for t_0 is reconstructed. We call this algorithm Algorithm B, and present a pseudocode of it in Appendix A.

Next we state a proposition that governs the optimality of Algorithm B for the UEP packetization problem in the convex case. For not interrupting the flow of our exposition we defer its proof to Appendix B.

Proposition 1. If the rate-fidelity function $\phi(r)$ is convex, then the L-slice redundancy assignment computed by Algorithm B satisfies the constraint $f_1 \ge f_2 \ge \cdots \ge f_L$.

Algorithm B computes $O(NL^2)$ instances of $C(k,t)$. If the computation of each value $C(k,t)$ is done by a full search, then it requires $O(N)$ time and the complexity of the algorithm becomes $O(N^2L^2)$. However, one can do much better. This complexity can be reduced to $O(NL^2)$ by applying an elegant matrix-search technique introduced by Aggarwal *et al.* [1]. (A detailed description of this technique can be found in [3]). For each k, $2 \le k \le L$, consider the upper triangular matrix G_k with the elements $G_k(j,t), 0 \le j \le t \le k(N-1)$, where $G_k(j,t)$ is defined by

$$G_k(j,t) = C(k-1,j) + c_N(t-j)(\phi(kN-t) - \phi((k-1)N-j)) \tag{14}$$

for $t - N + 1 \le j \le t$, and $G_k(j,t) = -\infty$ for $0 \le j < t - N + 1$. Then relation (13) becomes equivalent to

$$C(k,t) = \max_{0 \le j \le k(N-1)} G_k(j,t). \tag{15}$$

In other words, for a given k, the search required by (13) corresponds to finding the maximum element in the column t of G_k. Matrix G_k is said to be totally monotone with respect to column maxima if for $j < j'$ and $t < t'$, the following implication holds:

$$G_k(j',t) \ge G_k(j,t) \Rightarrow G_k(j',t') \ge G_k(j,t'). \tag{16}$$

As demonstrated by Aggarwal *et al.* in [1], all the m column maxima of an $m \times m$ matrix can be found in $O(m)$ time if the matrix is totally monotone. Very encouragingly, we can indeed show that the matrix G_k defined in (14) is totally monotone if $p_N(n)$ is monotonically non-increasing. This result is stated by the following proposition, whose proof is presented in Appendix B.

Proposition 2. If the packet loss probability $p_N(n)$ is decreasing in n, then the upper triangular matrix G_k is totally monotone with respect to column maxima.

The total monotonicity of G_k for non-increasing $p_N(n)$ enables us to apply the matrix-search technique of Aggarwal *et al.* [1] to compute all column maxima of G_k for given k, in $O(NL)$ time. Therefore, computing all values of $C(k,t)$ takes $O(NL^2)$ time. Consequently, the time complexity of optimal UEP packetization is reduced to $O(NL^2)$ when $\phi(r)$ is convex and $p_N(n)$ is non-increasing. In the case the maximum value of column t is not unique, the matrix search algorithm chooses the one of the largest row index $j = \jmath(k,t)$. This tie break rule is the same as in Algorithm B, which is also a condition that we need in the proof of Proposition 1.

To assess the space complexity, we note that $O(NL)$ entries of $C(k,t)$ have to be stored at any given time for applying the recursion (13). Moreover the algorithm needs to store $O(NL^2)$ entries of $\jmath(k,t)$ in order to reconstruct the optimal redundancy assignment. The latter space requirement dominates, hence the overall space complexity is $O(NL^2)$.

The fast matrix search algorithm can also be applied to a wider class of erasure channels that do not even have non-increasing $p_N(n)$, as long as the rate-fidelity function is convex. Consider an independent erasure channel with the packet loss rate ϵ. The probability of losing n packets out of N is $p_N(n) = \binom{N}{n}\epsilon^n(1-\epsilon)^{N-n}$. The probability mass function $p_N(n)$ is not monotone, but it is a unimodal function with its peak at $n_0 = \lfloor \epsilon(N+1) \rfloor$. We have $p_N(n) \leq p_N(n+1)$ for $n < n_0$ and $p_N(n) \geq p_N(n+1)$ for $n \geq n_0$. This result is stated (as Lemma 1) and proved in Appendix B. Since $p_N(n)$ is decreasing for $n \geq n_0$, the fast matrix search technique can still be applied, if the algorithm restricts to redundancy assignments with at least n_0 protection symbols to each slice. In other words we narrow the search range for j in (13) to

$$n_0 \leq t - j \leq N - 1. \tag{17}$$

According to Proposition 1, the assignment output by Algorithm B with this modification satisfies the constraint (1), too. Moreover, this assignment is still optimal for the UEP packetization problem, in spite of the restriction (17), if the packet loss rate ϵ is at most $\frac{N}{2(N+1)}$, as stated by the following proposition.

Proposition 3. If the rate-fidelity function $\phi(r)$ is convex and the channel is an independent erasure channel with packet loss rate $\epsilon \leq \frac{N}{2(N+1)}$, then there is an L-slice redundancy assignment maximizing (3) with at least n_0 protection symbols on each slice, where $n_0 = \lfloor \epsilon(N+1) \rfloor$.

We again defer the proof of this result to the appendix for not interrupting the flow of the presentation. Note that the assumption $\epsilon \leq \frac{N}{2(N+1)}$ is very reasonable in practice since the threshold $\frac{N}{2(N+1)}$ is very close to the value 0.5.

The fast matrix search algorithm can also be used as an approximation algorithm when the rate-fidelity function $\phi(r)$ is not convex. We only need a slight modification to ensure that the output L-slice redundancy assignment satisfies the constraint (1). The modification is made in the matrix G_k for all k. The value $G_k(j,t)$ is set to -1 if $t - j > j - \jmath(k-1, j)$. In other words, the value j will not be a candidate at the selection of $\jmath(k,t)$ if it introduces a reverse of the order in the final redundancy assignment. Note that this artificially imposed order may miss the globally optimal solution to the problem if $\phi(r)$ is not convex. But in practice, this seldom happens. Even if it does, the loss of optimality by using matrix search instead of the exact Algorithm A is negligible as we will see in Section 5.

The algorithms proposed by Mohr, Ladner, and Riskin [8] and by Puri and Ramchandran [9] work on convex hull of operational rate-fidelity function $\phi(r)$ of the source sequence, and both can obtain optimal solution to the UEP packetization problem, if $\phi(r)$ is convex and if fractional redundancy bit allocation is allowed. Under the practical constraint of integer redundancy assignment, however, these two algorithms are still suboptimal even if $\phi(r)$ is strictly convex. The algorithm in [8] has the complexity $O(hN \log N)$, where h is the number of points on the convex hull of the rate-fidelity function. If $\phi(r)$ is convex, then its complexity becomes $O(N^2 L \log N)$, which is asymptotically higher than our $O(NL^2)$ algorithm, since typically $N \log N > L$ for reasonably long scalable sequences. The algorithm in [9] uses a Lagrangian multiplier λ. The algorithm is linear in the number of packets (N) for a given λ, and the value of λ to meet the target rate is found using a fast bisection search (it was usually found within 32 iterations according to [10]). The algorithm of Stankovic, Hamzaoui, and Xiong [14] is an $O(NL)$ time local search algorithm that starts from a solution that maximizes the expected number of source bits and iteratively improves the solution. It is the fastest among all UEP algorithms and offers very good approximation solution in practice.

4 Optimization of the Product Code

The algorithms proposed above can also be used to optimize the product code introduced by Sachs, Anand, and Ramchandran in [10], for the case when both packet loss and bit errors within packets can happen simultaneously. The idea of [10] is to use Reed-Solomon block codes of different strengths to protect the subsequent source slice as described in Section 1, and then to additionally protect all columns with the same RCPC+CRC code. The column code provides protection against bit errors inside each packet. If the length of a packet is L then the number of slices containing source and RS symbols is $L(r) = \lfloor rL \rfloor$, where r is the rate of the RCPC+CRC column code. For a given rate r, the problem of maximizing the expected fidelity of the product code is equivalent to the optimal UEP packetization problem as defined in Section 1 but with some changed parameters. Namely, L and $p_N(n)$ have to be replaced by $L(r)$, respectively, $p_{N,r}(n)$, where $p_{N,r}(n)$ is the probability that exactly n packets out of N can not be recovered by the receiver (i.e. they are either lost or corrupted after RCPC decoding). This probability depends on both packet erasure statistics and on error correction capability of the RCPC+CRC column code used to protect data inside each packet. For instance, if the channel is a concatenation of a BSC (binary symmetric channel) and an independent packet erasure channel with packet erasure rate ϵ, then $p_{N,r}(n)$ can be evaluated as in [15]:

$$p_{N,r}(n) = \binom{N}{n} \epsilon'^n (1 - \epsilon')^{N-n}, \tag{18}$$

where $\epsilon' = \epsilon + (1 - \epsilon)q(r)$ and $q(r)$ is the probability that a packet can not be correctly decoded with the RCPC decoder.

The product code of UEP packetization and RCPC+CRC can be optimized by solving the optimal UEP packetization problem for each possible rate r of the column code, and selecting the rate r_0 corresponding to the largest expected fidelity. Consequently, if Algorithm A is used for computing the optimal assignment of RS redundancy for each rate r, the globally optimal product code can be obtained. Similarly, the use of matrix search algorithm achieves global optimality for convex rate-fidelity functions and provides approximate solutions in general case.

Inputs	Alg. [8]	Alg. A	Alg. B
barboon	20.7643	20.7687	20.7687
zelda	32.7260	32.7432	32.7417
lena	29.4156	29.4360	29.4358
convex	28.0874	28.1060	28.1060
concave	18.8518	22.1725	19.9004

Table 1: The expected PSNR in dB of different UEP packetization algorithms for an exponential packet loss probability function.

5 Experimental Results

We implemented Algorithms A and B, and tested them on various scalable source sequences of different operational rate-fidelity curves. The results of our algorithms were compared with those of other algorithms in the literature. Table 1 presents average PSNR results of our algorithms and the algorithm of [8]. The average was computed using (3) for a packet erasure channel whose packet loss probability $p_N(n)$ is exponentially decreasing in n with a mean loss rate of 0.2. Note that this is the same experimental condition as in [8]. In Table 1 we compare the three algorithms on five scalable source sequences. Three of the sequences "barboon", "zelda" and "lena" are real embedded data streams of well-known test images compressed by SPIHT [11]. The sequences "convex" and "concave" are synthesized data generated to evaluate the performance of different algorithms in extreme cases. The input sequence "convex" has a strictly convex rate-fidelity function $\phi(r)$ (i.e., every operational point of $\phi(r)$ is on the convex hull), whereas the input sequence "concave" has a rate-fidelity function $\phi(r)$ consisting of three concatenated concave pieces (i.e., the convex hull of $\phi(r)$ has only four vertices). In the experiments we allocated a transmission budget of 147 packets with 48 symbols per packet, where each symbol consists of 8 bits.

One can observe from Table 1 that Algorithm A outperforms the other two algorithms on all test data as it should, although the margin is quite slim on SPIHT scalable code streams of natural images. If the rate-fidelity function $\phi(r)$ is strictly convex, Algorithm B and Algorithm A produce

N	Alg. A	Alg. B	Alg. [7]	Alg. [8]	Alg. [9]	Alg. [14]
100	28.27	28.27	28.26	28.26	28.24	28.26
200	30.99	30.99	30.95	30.97	30.95	30.93

Table 2: The expected PSNR in dB of different UEP packetization algorithms on SPIHT bitstream of Lenna image, transmitted over a packet erasure channel with an exponentially decreasing probability function $p_N(n)$, with mean loss rate 0.2.

identical redundancy assignment and hence the same PSNR, verifying our theoretical result that Algorithm B is globally optimal for convex $\phi(r)$. We point out, for the sake of completeness, that our globally optimal algorithm can outperform the algorithm of [8] by as much as 3.3 dB if $\phi(r)$ is highly non-convex. Also, in this case, the suboptimality of Algorithm B is clearly exhibited with a PSNR value 2.27 dB lower than the optimal. However, Algorithm B is still superior to the algorithm of [8] by a margin of 1.05 dB.

For more comprehensive comparisons, we also provided Stankovic, Hamzaoui, and Xiong [14, 15] with the source programs that implemented our algorithms. They conducted a thorough empirical study on all existing UEP algorithms including their own. On their courtesy we present some of their relevant findings. In this experiments the allocated transmission budget is of 100 or 200 packets with 48 symbols per packet, where each symbol consists of 8 bits.

Tables 2 and 3 tabulate the expected PSNR of our algorithms in comparison with the algorithms of [7, 8, 9, 14] on SPIHT and JPEG 2000 bitstreams of Lenna image, respectively. The experiments were carried out for a simulated packet erasure channel whose packet loss probability function $p_N(n)$ was exponentially decreasing with mean loss rate 0.2.

Table 4 presents the expected PSNR of the SPIHT bitstream of Lenna image for a two-state packet loss model where the average packet loss probability is 0.1 and the average length of burst errors is 9.57. The probability function $p_N(n)$ is computed with the method of [6].

Tables 5, 6 and 7 show the results for the optimization of the product code [10]. A 16-bit CRC code with generator polynomial 0×15935 is used. The generator polynomials of the RCPC codes are $(0117, 0127, 0155, 0171)$, the mother code rate is $1/4$ and the puncturing rate is 8. Thus the

N	Alg. A	Alg. B	Alg. [7]	Alg. [8]	Alg. [9]	Alg. [14]
100	28.00	27.72	27.13	27.98	27.80	27.94
200	30.80	30.77	27.77	30.79	30.74	30.76

Table 3: The expected PSNR in dB of different UEP packetization algorithms on JPEG 2000 bitstream of Lenna image transmitted over a packet erasure channel with an exponentially decreasing probability function $p_N(n)$, with mean loss rate 0.2.

N	Alg. A	Alg. B	Alg. [7]	Alg. [8]	Alg. [9]	Alg. [14]
100	29.68	29.68	29.65	29.67	29.66	29.67
200	33.03	33.03	33.00	33.03	33.03	33.02

Table 4: The expected PSNR in dB of different UEP packetization algorithms on SPIHT bitstream of Lenna image, for a two-state packet loss model with average packet loss probability 0.1 and average length of burst errors 9.57.

set of RCPC rates is $\{8/32, \cdots, 8/9\}$. The strategy used for optimization is the one described in Section 4, namely, the optimal, respectively approximately optimal, RS redundancy assignment for each RCPC code rate r is computed by using, respectively, our algorithms A, B and the algorithms of [8] and [9]. Then the code rate which yields the maximum expected PSNR is selected. Also the results obtained by the local search strategy of [15] are presented. The results of Tables 5 and 6 are for a concatenation between a packet erasure channel with packet loss rate 0.05 and a BSC (binary symmetric channel) with bit error rate 0.1. Table 5 presents the test results for the SPIHT bitstream of Lenna image, and Table 6 for the JPEG 2000 bitstream of the same image. The test results of Table 7 are for the SPIHT bitstream and for a Rayleigh fading channel with SNR=10 dB and $f_D = 10^{-5}$.

In summary, for most scalable source sequences in practice the difference in rate-fidelity performance between our UEP packetization algorithms and others is quite small. But at least theoretically this difference can be large if the rate-fidelity function is highly non-convex. This paper offers exact solutions to the problem at reduced complexity than the previously known globally optimal algo-

N	Alg. A	Alg. B	Alg. [8]	Alg. [9]	Alg. [15]
100	26.16	26.14	26.14	26.15	26.14
200	28.53	28.53	28.50	28.51	28.51

Table 5: The expected PSNR in dB of the product code of intra-packet RCPC+CRC and UEP packetization optimized by different algorithms for the SPIHT bitstream of Lenna image transmitted over a packet erasure channel with packet loss rate 0.05, concatenated with a BSC with bit error rate 0.1.

N	Alg. A	Alg. B	Alg. [8]	Alg. [9]	Alg. [15]
100	25.80	25.53	25.77	25.78	25.74
200	28.27	28.27	28.14	28.21	28.18

Table 6: The expected PSNR in dB of the product code of intra-packet RCPC+CRC and UEP packetization optimized by different algorithms for the JPEG 2000 bitstream of Lenna image transmitted over a packet erasure channel with packet loss rate 0.05, concatenated with a BSC with bit error rate 0.1.

N	Alg. A	Alg. B	Alg. [8]	Alg. [9]	Alg. [15]
200	31.01	31.01	30.98	31.00	30.95

Table 7: The expected PSNR in dB of the product code of intra-packet RCPC+CRC and UEP packetization optimized by different algorithms for the SPIHT bitstream of Lenna image transmitted over a Rayleigh fading channel with SNR=10 dB and $f_D = 10^{-5}$.

rithm [10]. More importantly, it constructively quantifies the efficacy of other fast, good, and more practical approximation algorithms [7, 8, 9, 14].

6 Conclusion

In this paper we investigated discrete optimization approach to uneven error-protected packetization of scalable code streams, an important problem in multimedia streaming over noisy channels. Two new algorithms were proposed to compute globally optimal solutions of the problem, one for the general case of arbitrary source and channel statistics, and the other for convex rate-fidelity functions. These new algorithms have lower asymptotical complexity or/and fewer constraints than the algorithms of [7, 8, 9, 10, 16].

Appendix A

Algorithm A. Optimal UEP Packetization for General Case.

```
for     n := 0 to N − 1 do
        A(0, n, 0) := 0
end for
for     k := 1 to L do
        for     n := N − 1 downto 0 do
                for     t := kn to k(N − 1) do
                        Z(k, n, t) := 0;
                        if      t ≤ n + (k − 1)(N − 1) then
                                A(k, n, t) := A(k − 1, n, t − n) + c_N(n)(φ(kN − t) − φ((k − 1)N − t + n));
                                if      t ≥ k(n + 1) then
                                        if      A(k, n, t) < A(k, n + 1, t) then
                                                A(k, n, t) := A(k, n + 1, t); Z(k, n, t) := 1
                                        end if
                                end if
                        else
                                A(k, n, t) := A(k, n + 1, t); Z(k, n, t) := 1
                        end if
                end for
        end for
end for
t_0 := max^{-1}_{0≤t≤L(N−1)} A(L, 0, t);
n := 0; t := t_0;
for     k := L downto 1 do
        j := n;
        while   Z(k, n, t) = 1 do
                j := j + 1
        end while
        f_k := j; t := t − j; n := j
end for
```

Algorithm B. UEP packetization without constraint of decreasing redundancy assignment.

```
for     t := 0 to N − 1 do
        C(1,t) := Φ_1(t); ȷ(1,t) := 0
end for
for     k := 2 to L do
        for     t := 0 to k(N − 1) do
                C(k,t) := max_{j,0≤t−j≤N−1}{C(k − 1,j) + c_N(t − j)(φ(kN − t) − φ((k − 1)N − j))};
                ȷ(k,t) := max{max^{−1}_{j,0≤t−j≤N−1}{C(k − 1,j) + c_N(t − j)(φ(kN − t) − φ((k − 1)N − j))}}
        end for
end for
t_0 := max^{−1}_{0≤t≤L(N−1)} C(L,t)
f_L := t_0 − ȷ(L,t_0); t := t_0 − f_L
for     j := 1 to L − 1 do
        k := L − j; f_k := t − ȷ(k,t); t := t − f_k
end for
```

Appendix B

Proposition 1. If the rate-fidelity function $\phi(r)$ is convex, then the L-slice redundancy assignment computed by Algorithm B satisfies the constraint (1) (i.e. $f_1 \geq f_2 \geq \cdots \geq f_L$).

Proof. Let $F = (f_1, \cdots, f_L)$ be the L-slice redundancy assignment computed by Algorithm B. Assume that there is some $k, 1 \leq k \leq L-1$, such that

$$f_k < f_{k+1}. \tag{19}$$

Let F' be the L-slice redundancy assignment obtained from F by interchanging the redundancy assignments for the k-th and the $(k+1)$-th slice. We will show that

$$\Phi_L(F') > \Phi_L(F), \tag{20}$$

which contradicts the optimality of Algorithm B for the unconstrained version of the optimal UEP packetization problem. For this note first that the two assignments F and F' incur the same disposition of the source symbols on all the slices excepting the k-th and the $(k+1)$-th slices. Hence only the contribution of the k-th and the $(k+1)$-th slice to the expected fidelity differ for the two assignments. It follows that the relation (20) we aim to prove, is equivalent to

$$\begin{aligned} &c_N(f_{k+1})(\phi(r'_k) - \phi(r_{k-1})) + c_N(f_k)(\phi(r_{k+1}) - \phi(r'_k)) > \\ &\quad c_N(f_k)(\phi(r_k) - \phi(r_{k-1})) + c_N(f_{k+1})(\phi(r_{k+1}) - \phi(r_k)), \end{aligned} \tag{21}$$

where r'_k denotes the number of source symbols on the first k-th slices according to the assignment F'. The above inequality is further equivalent to

$$(c_N(f_{k+1}) - c_N(f_k))(\phi(r'_k) - \phi(r_{k-1}) - \phi(r_{k+1}) + \phi(r_k)) > 0. \tag{22}$$

Since $r'_k - r_{k-1} = r_{k+1} - r_k = N - f_{k+1}$ and $r_{k-1} < r_k$, the convexity of the function $\phi(r)$ implies that

$$\phi(r'_k) - \phi(r_{k-1}) \geq \phi(r_{k+1}) - \phi(r_k), \tag{23}$$

which, together with the relation $c_N(f_{k+1}) > c_N(f_k)$, leads to the conclusion

$$(c_N(f_{k+1}) - c_N(f_k))(\phi(r'_k) - \phi(r_{k-1}) - \phi(r_{k+1}) + \phi(r_k)) \geq 0. \tag{24}$$

We have to show now that the above relation cannot hold with equality. Indeed, the equality in relation (24) implies that the $(k+1)$-slice redundancy assignment $(f_1, \cdots f_{k-1}, f_{k+1}, f_k)$ achieves the same value for the expected fidelity for the first $k+1$ slices as $(f_1, \cdots f_{k-1}, f_k, f_{k+1})$ and this value is the maximal one corresponding to the first $k+1$ slices and $t = f_1 + \cdots + f_{k-1} + f_k + f_{k+1}$ protection symbols (according to the algorithm):

$$\Phi_{k+1}(f_1, \cdots, f_{k-1}, f_k, f_{k+1}) = \Phi_{k+1}(f_1, \cdots, f_{k-1}, f_{k+1}, f_k) = C(k, t). \tag{25}$$

When such a situation occurs the algorithm chooses the assignment with the smallest number of protection symbols on the $(k+1)$-th slice. Since $(f_1, \cdots, f_k, f_{k+1})$ is the redundancy assignment chosen by the algorithm, it follows that $f_{k+1} \leq f_k$, which contradicts the inequality (19).□

Proposition 2. If the packet loss probability $p_N(n)$ is decreasing in n, then the upper triangular matrix G_k is totally monotone with respect to column maxima.

Proof. If $t' - j > N - 1$, then $G_k(j, t') = -\infty$ and relation (16) trivially holds.

Suppose now that $t' - j \leq N - 1$. Since $j < j' \leq t < t'$, it follows that

$$N - 1 \geq t' - j > t - j > t - j' \geq 0, \tag{26}$$

$$N - 1 \geq t' - j > t' - j' > t - j' \geq 0. \tag{27}$$

Hence, all quantities appearing in (16) can not be $-\infty$. For proving (16), it is enough to show that

$$G_k(j, t) - G_k(j', t) \geq G_k(j, t') - G_k(j', t') \tag{28}$$

Denote $x = \phi(kN-t)$, $y = \phi((k-1)N-j)$, $u = \phi(kN-t')$, $v = \phi((k-1)N-j')$, $\alpha = c_N(t-j)-c_N(t-j')$, $\beta = c_N(t'-j)-c_N(t-j)$, $\gamma = c_N(t'-j)-c_N(t'-j')$, $\delta = c_N(t'-j')-c_N(t-j')$. Thus the inequality (28) is equivalent to

$$\alpha x + \beta y \geq \gamma u + \delta v. \tag{29}$$

Inequalities (26) and(27) imply that $kN-t > (k-1)N-j > (k-1)N-j'$ and $kN-t > kN-t'$. Since the rate-fidelity function $\phi(r)$ is increasing, it follows that $x \geq y \geq v$ and $x \geq u$. Also, the function $c_N(n)$ is increasing, being a cumulative distribution function. Thus we conclude, aided again by inequalities (26) and (27), that the values α, β, γ, and δ are nonnegative.

Note that $\beta = \sum_{i=1}^{t'-t} p_N(t-j+i)$ and $\delta = \sum_{i=1}^{t'-t} p_N(t-j'+i)$. Since $t-j > t-j'$ and $p_N(n)$ is decreasing, it follows that $p_N(t-j+i) \leq p_N(t-j'+i)$ for all $1 \leq i \leq t'-t$. In conclusion, $\beta \leq \delta$.

Furthermore, since $x \geq y$ and $\delta - \beta \geq 0$, we obtain $(\delta-\beta)x \geq (\delta-\beta)y$. Using the equality $\alpha-\gamma = \delta-\beta$, the relation $(\alpha-\gamma)x \geq (\delta-\beta)y$ follows, which is equivalent to $\alpha x+\beta y \geq \gamma x+\delta y$. Also the inequalities $x \geq u$, $y \geq v$, together with $\gamma, \delta \geq 0$ lead to the conclusion that $\gamma x + \delta y \geq \gamma u + \delta v$. Now the validity of equation (29) is established. □

Lemma 1. Let ϵ be a positive value smaller than 1 and $p_N(n) = \binom{N}{n}\epsilon^n(1-\epsilon)^{N-n}$ for all $n, 0 \leq n \leq N$. Also let $n_0 = \lfloor \epsilon(N+1) \rfloor$. Then $p_N(n) \leq p_N(n+1)$ for all $n, 0 \leq n < n_0$ and $p_N(n) \geq p_N(n+1)$ for all $n, n_0 \leq n < N$.

Proof. By straightforward algebraic computations the equivalence between the relations $p_N(n) \leq p_N(n+1)$ and $n \leq (N+1)\epsilon - 1$ is obtained. If $n < n_0$ it follows that $n \leq (N+1)\epsilon - 1$, hence $p_N(n) \leq p_N(n+1)$. If $n \geq n_0$, then $n > (N+1)\epsilon - 1$, hence $p_N(n) > p_N(n+1)$. □

Proposition 3. If the rate-fidelity function $\phi(r)$ is convex and the channel is an independent erasure channel with packet loss rate $\epsilon \leq \frac{N}{2(N+1)}$, then there is an L-slice redundancy assignment maximizing (3) with at least n_0 protection symbols on each slice, where $n_0 = \lfloor \epsilon(N+1) \rfloor$.

Proof. It is enough to show that if an L-slice redundancy assignment $F = (f_1, \cdots, f_L)$ has $f_k < n_0$ for some $k, 1 \leq k \leq L$, then the assignment F' obtained by replacing f_k with n_0 yields an expected fidelity no smaller. For each slice $i, 1 \leq i \leq L$ denote by r_i, respectively r'_i, the number of source symbols situated on the first i slices according to the assignment F, respectively F'. Note first that the two assignments F and F' imply the same disposition of the source symbols on the first $k-1$ slices, hence the contribution to the expected fidelity due to the first $k-1$ slices is the same for the two assignments. Also note that for each slice $i, i > k$, we have $r_i > r'_i$ and $r_i - r_{i-1} = r'_i - r'_{i-1} = N - f_i$. Then the convexity of the function $\phi(r)$ implies that

$$\phi(r_i) - \phi(r_{i-1}) \leq \phi(r'_i) - \phi(r'_{i-1}). \tag{30}$$

Since both F and F' provide the same protection to each slice $i, i > k$, we conclude using the above relation, that the contribution to the expected fidelity of the slices from $k+1$ to L is smaller or equal for assignment F than for assignment F'.

To complete the proof it is now sufficient to show that the contribution of the k-th slice to the expected fidelity for assignment F is at most equal to that for assignment F', i.e.

$$c_N(f_k)(\phi(r_k) - \phi(r_{k-1})) \leq c_N(n_0)(\phi(r'_k) - \phi(r'_{k-1})). \tag{31}$$

Since $r_k - r_{k-1} = N - f_k$ and $r'_k - r'_{k-1} = N - n_0$, the above relation is equivalent to

$$c_N(f_k)(N - f_k)\frac{\phi(r_k) - \phi(r_{k-1})}{r_k - r_{k-1}} \leq c_N(n_0)(N - n_0)\frac{\phi(r'_k) - \phi(r'_{k-1})}{r'_k - r'_{k-1}}. \tag{32}$$

Because the function $\phi(r)$ is convex and non-decreasing, the relations $r_{k-1} = r'_{k-1}$, $r_k - r_{k-1} > r'_k - r'_{k-1}$, imply that

$$0 \leq \frac{\phi(r_k) - \phi(r_{k-1})}{r_k - r_{k-1}} \leq \frac{\phi(r'_k) - \phi(r'_{k-1})}{r'_k - r'_{k-1}}. \tag{33}$$

Consequently, for proving (32) it is sufficient to additionally show that

$$c_N(f_k)(N - f_k) \leq c_N(n_0)(N - n_0). \tag{34}$$

This inequality is equivalent to

$$(N - f_k)\sum_{i=0}^{f_k} p_N(i) \leq (N - n_0)\sum_{i=0}^{n_0} p_N(i), \tag{35}$$

further equivalent to

$$(n_0 - f_k)\sum_{i=0}^{f_k} p_N(i) \leq (N - n_0)\sum_{i=f_k+1}^{n_0} p_N(i). \tag{36}$$

It is very easy to check that for $i < n_0$ we have $p_N(i) \leq p_N(i+1)$. This implies that the left hand expression in the above inequality is at most equal to $(n_0 - f_k)(f_k + 1)p_N(f_k + 1)$, while the right hand expression is larger or equal than $(N - n_0)(n_0 - f_k)p_N(f_k + 1)$. Hence for proving (36) it is enough to show that

$$(n_0 - f_k)(f_k + 1)p_N(f_k + 1) \leq (N - n_0)(n_0 - f_k)p_N(f_k + 1). \tag{37}$$

The condition in the hypothesis that $\epsilon \leq \frac{N}{2(N+1)}$ implies that $n_0 \leq \frac{N}{2}$, hence $N - n_0 \geq n_0 \geq f_k + 1$, which yields the inequality (37) thus completing the proof. □

References

[1] A. Aggarwal, M. Klave, S. Moran, P. Shor, and R. Wilber, "Geometric applications of a matrix-searching algorithm", *Algorithmica*, 2(1987), pp. 195-208.

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

[3] A. Apostolico and Z. Galil(eds.), *Pattern Matching Algorithms*, New York 1997.

[4] V. Chande and N. Farvardin, "Joint source-channel coding for progressive transmission of embedded source coders", *Proc. DCC'99 Data Compression Conference*, pp. 52-61, March 1999.

[5] G. Davis, J. Danskin, "Joint source and channel coding for image transmission over lossy packet networks", *SPIE Conf. Wavelet Applications of Digital Image Proc. XIX* , pp. 376-387, Denver, August 1996.

[6] U. Horn, K. Stuhlmuller, M. Link, and B. Girod, " Robust internet video transmission based on scalable coding and unequal error protection", *Image Communication (Special Issue on Real-time Video over the Internet)*, vol.15, no. 1-2, pp. 77-94, Sept. 1999.

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

[8] A. E. Mohr, R. E. Ladner, and E.A. Riskin, " Approximately optimal assignment for unequal loss protection", in *Proc. ICIP 2000 International Conf. on Image Processing*, vol. 1, pp. 367-370, Sept. 2000.

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

[10] D. G. Sachs, R. Anand, and K. Ramchandran, "Wireless image transmission using multiple-description based concatenated codes", *Proc. SPIE 2000*, vol. 3974, pp. 300-311, Jan. 2000.

[11] A. Said and W. A. Pearlman, "A new, fast, and efficient image codec based on set partitioning in hierarchical trees", *IEEE Trans. Circuits and Systems for Video Tech.*, vol. 6, pp. 243-250, June 1996.

[12] P.G. Sherwood and K. Zeger, "Progressive image coding on noisy channels", *Proc. DCC'97*, pp. 72-79, March 1997.

[13] V. Stankovic, R. Hamzaoui, and D. Saupe, "Fast algorithm for optimal error protection of embedded wavelet codes", *Proc. MMSP-01 IEEE Workshop on Multimedia Signal Processing*, Cannes, Oct. 2001.

[14] V. Stankovic, R. Hamzaoui, and Z. Xiong, "Packet loss protection of embedded data with fast local search", *Proc. ICIP'2002* , vol. 2, pp. II-165–II-168, Sept. 2002.

[15] V. Stankovic, R. Hamzaoui, and Z. Xiong, "Joint product code optimization for scalable multimedia transmission over wireless channels", *Proc. ICME'2002* , vol. 1, pp. 865-868, Aug. 2002.

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

[17] D. Taubman, "High performance scalable image compression with EBCOT," *IEEE Trans. Image Processing*, vol. 9, pp. 1158-1170, July 2000.