

# Scalable Video Multicast Using Expanding Window Fountain Codes

Dejan Vukobratović, *Member, IEEE*, Vladimir Stanković, *Member, IEEE*, Dino Sejdinović, *Student Member, IEEE*, Lina Stanković, *Member, IEEE*, and Zixiang Xiong, *Fellow, IEEE*

**Abstract**—Fountain codes were introduced as an efficient and universal forward error correction (FEC) solution for data multicast over lossy packet networks. They have recently been proposed for large scale multimedia content delivery in practical multimedia distribution systems. However, standard fountain codes, such as LT or Raptor codes, are not designed to meet unequal error protection (UEP) requirements typical in real-time scalable video multicast applications. In this paper, we propose recently introduced UEP expanding window fountain (EWF) codes as a flexible and efficient solution for real-time scalable video multicast. We demonstrate that the design flexibility and UEP performance make EWF codes ideally suited for this scenario, i.e., EWF codes offer a number of design parameters to be “tuned” at the server side to meet the different reception criteria of heterogeneous receivers. The performance analysis using both analytical results and simulation experiments of H.264 scalable video coding (SVC) multicast to heterogeneous receiver classes confirms the flexibility and efficiency of the proposed EWF-based FEC solution.

**Index Terms**—Fountain codes, H264 SVC, scalable video multicast, unequal error protection.

## I. INTRODUCTION

**E**FFICIENT multicast transmission of scalable video content over lossy packet networks to heterogeneous receivers is still a challenge. Scalable video coding techniques enable the receivers to progressively improve reconstructed video quality with the amount of the data received. This may

Manuscript received October 01, 2008; revised April 07, 2009. Current version published September 16, 2009. The work of D. Vukobratović was supported by a Marie Curie Intra European Fellowship within the 7th European Community Framework Programme. The material in this paper was presented in part at the 45th Annual Allerton 2007 Conference on Communications, Computing and Control, Monticello, IL, September 26–28, 2007, and in part at the IEEE International Conference on Multimedia and Expo ICME 2008, Hannover, Germany, June 23–26, 2008. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Aggelos K. Katsaggelos.

D. Vukobratović is with the Department of Power, Electronics and Communication Engineering, University of Novi Sad, 21000 Novi Sad, Serbia, and also with the Department of Electronic and Electrical Engineering, University of Strathclyde, Glasgow G1 1XW, U.K. (e-mail: dejanv@uns.ac.rs).

V. Stanković and L. Stanković are with the Department of Electronic and Electrical Engineering, University of Strathclyde, Glasgow G1 1XW, U.K. (e-mail: vladimir.stankovic@eee.strath.ac.uk; lina.stankovic@eee.strath.ac.uk).

D. Sejdinović is with the Centre for Communications Research, Department of Electrical and Electronic Engineering, University of Bristol, Bristol BS8 1UB, U.K. (e-mail: d.sejdinovic@bristol.ac.uk).

Z. Xiong is with the Department of Electrical and Computer Engineering, Texas A&M University, College Station, TX 77843 USA (e-mail: zx@ece.tamu.edu).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TMM.2009.2026087

enable receivers with increased capabilities (available bandwidth, screen resolution, processing power, etc.) to experience better video quality, while at the same time providing the basic reconstruction quality for low capability receivers. However, even for high capability receivers, packet losses in scalable video transmission may significantly deteriorate the quality of the reconstructed data. For example, an early packet loss in the transmission of a typical scalable coded data segment, where the data importance decreases along the data segment, may lead to severe error propagation. For this reason, the state-of-the-art real-time scalable video distribution systems rely on powerful forward error correction (FEC) mechanisms at the video server side, optimized with respect to the scalable source output bitstreams (see [1] for an overview of this topic).

Traditionally, Reed–Solomon (RS) codes [2] are applied as the FEC solution for error-resilient multimedia multicast over networks with packet erasures. Examples are layered hybrid automatic repeat request (ARQ) and FEC schemes (HARQ/FEC) investigated in [3] and [4]. Recently, fountain codes [5], such as LT codes [6] or Raptor codes [7], have been proposed as a more flexible and efficient solution for scalable data multicast over lossy packet networks [8]–[12]. Raptor codes provide linear encoding/decoding complexity and universal capacity-approaching behavior for any channel packet loss rate at the price of a small reception overhead, as compared to RS codes. These codes have been proposed as application-layer forward error correcting (AL-FEC) solution for large scale multimedia content delivery in practical systems. Due to their advantages in terms of complexity, performance and flexibility over RS codes applied at the link layer [13], rateless AL-FEC solutions became part of recent standardization efforts in systems such as multimedia broadcast multicast services (MBMS) within 3GPP UMTS networks [14] and IP-Datcast (IPDC) within DVB-H networks [15].

For delay-constrained applications, such as real-time video streaming, the fountain encoder cannot make use of its “rateless” property as proposed in the original framework [5]. Indeed, the encoder can produce “only” a finite amount of encoded symbols per source block before moving to the next source block. In this scenario, many receivers might not be able to collect enough encoded symbols to perform successful decoding of the source block. In addition, scalable sources do not require that each receiver recovers the entire source block, but as many input symbols as possible from its beginning onwards, because recovering additional symbols progressively increases reconstruction quality. Therefore, we can identify two major drawbacks of standard fountain solutions for scalable video multicast applications, namely: 1) if a minimum amount of encoded data

is not received, the iterative decoder can reconstruct a negligible portion of the transmitted video block, 2) standard fountain codes are equal error protection (EEP) codes, whereas scalable video transmission calls for unequal error protection (UEP) FEC schemes due to the unequal importance of data in the scalable bitstream. We note that recently proposed rateless solutions for scalable video streaming address these problems by associating separate rateless codes for each layer of scalable coded content, thereby increasing system complexity and achieving UEP performance by applying rather complex rate allocation algorithms for constituent rateless codes [8], [9], [12].

In this paper, we propose a solution for scalable video multicast based on UEP fountain codes named expanding window fountain (EWF) codes [16], [17] that addresses both of the aforementioned problems. The main advantage of EWF-based approach is that it is a “single code” solution with UEP performance which is analytically predictable using simple asymptotic analysis [17]. The design flexibility and UEP performance make EWF codes ideally suited for this scenario as they offer a number of design parameters to be “tuned” at the server side to simultaneously satisfy reception conditions of heterogeneous receivers by exploiting the scalable features of the video codec. Following the introductory section, in Section II we provide a short review of EWF codes and related analytical tools for predicting their performance. In Section III, our scalable EWF multicast system setting and EWF codes design methodology are presented, supported by numerical examples of code design optimization. In Section IV, we focus on EWF code optimization for scalable EWF multicast of scalable coded video stream encoded by recently introduced extension of H.264 AVC video coding standard [18] called H.264 scalable video coding (SVC) [19]. Both analytical results and simulation study are provided, confirming the flexibility and efficiency of the proposed EWF-based FEC solution, and excellent matching between theoretical predictions and simulation results. The paper is concluded in Section V.

## II. EXPANDING WINDOW FOUNTAIN (EWF) CODES

Digital fountain framework [5] is a universal capacity-approaching FEC solution for multicasting data over lossy packet networks. Its enabling component, fountain codes, are designed with inherent “rateless” property, that is, fountain codes may potentially produce an infinite stream of encoded symbols given the input symbols of a finite source block. The framework became a practical solution upon the introduction of LT codes [6]. LT codes provide a complete recovery of the transmitted source block, with high probability, for each receiver collecting any set of encoded symbols of size slightly larger than the number of input symbols. Their encoding/decoding complexity is of the order  $O(k \log k)$  for a source block of length  $k$ , due to the average degree of robust soliton degree distribution that scales logarithmically in  $k$  [6]. Raptor codes [7], an improvement over LT codes, are obtained by precoding LT codes defined by degree distribution of constant average value with high-rate low-density parity-check (LDPC) codes [21]. They represent a state-of-

the-art fountain solution with excellent performance and an encoding/decoding complexity of the order  $O(k)$ . However, standard LT and Raptor codes are EEP codes, because they place equal protection on each input symbol from the source block. Recently, fountain code designs with the UEP property have emerged [16], [17], [20]. In this section, we shortly review EWF codes [16], [17] as a basis for the proposed scalable multicast solution.

EWF codes are a novel class of UEP fountain codes based on the idea of “windowing” the source block to be transmitted. We assume that, for video streaming applications, EWF codes are applied on consecutive source blocks of  $k$  symbols (data packets). The set of expanding windows defined over the source block determines the set of importance classes associated with different quality layers of scalable coded video. For each importance class, asymptotic probability (as the source block length tends to infinity) that a symbol of the class is not recovered after  $l$  iterations of the iterative belief propagation (BP) decoder can be determined analytically using simple set of recursive formulae [17] that we review in this section. This analytical tool is a basis for the optimized EWF code design for scalable video transmission.

The sequence of  $r$  expanding windows, where each window is contained in the next window in the sequence, is defined over the source block (Fig. 1). The number  $r$  of expanding windows is equal to the number of importance classes of the source block. We denote the size of the  $i$ th window as  $k_i$ , where  $k_1 < \dots < k_r = k$ . The most important symbol class of size  $s_1 = k_1$  symbols is defined by the “innermost” window, and is protected by all the other windows in the sequence. The  $i$ th importance class,  $i \geq 2$ , of size  $s_i = k_i - k_{i-1}$  is the set of all input symbols that belong to the  $i$ th window, but do not belong to the  $(i-1)$ st window. The last  $r$ th window contains all the  $k$  symbols of the source block. We compactly describe the division of the source block into importance classes using polynomial notation as  $\Pi(x) = \sum_{i=1}^r \Pi_i x^i$ , where  $\Pi_i = (s_i/k)$ . In addition, it is useful to introduce  $\Theta_i = (k_i/k) = \sum_{j=1}^i \Pi_j$  to our notation. The set of expanding windows is characterized by a window selection probability distribution described by polynomial  $\Gamma(x) = \sum_{i=1}^r \Gamma_i x^i$ , where  $\Gamma_i$  is the probability of selecting the  $i$ th window. Finally, a degree distribution  $\Omega^{(j)}(x) = \sum_{i=1}^{k_j} \Omega_i^{(j)} x^i$  is associated with the  $j$ th expanding window,  $1 \leq j \leq r$ , and describes the LT encoding process performed over the data contained in that window.

EWF encoding proceeds in a slightly different fashion than the usual LT encoding. To create a new EWF encoded symbol, one of the windows is randomly selected with respect to the window selection probability distribution  $\Gamma(x)$ . Upon the window selection, a new encoded symbol is determined with an LT code described by the selected window degree distribution as if encoding were performed only on the input symbols from the selected window. This procedure is repeated at the EWF encoder for each encoded symbol. More formally, the EWF code  $\mathcal{F}_{EW}(\Pi, \Gamma, \Omega^{(1)}, \dots, \Omega^{(r)})$  assigns each encoded symbol to the  $j$ th window of size  $k_j$  with probability  $\Gamma_j$  and encodes the data from the selected window using the LT code with the degree distribution  $\Omega^{(j)}(x) = \sum_{i=1}^{k_j} \Omega_i^{(j)} x^i$ . Obviously, EWF code design generalizes the standard LT code design as LT

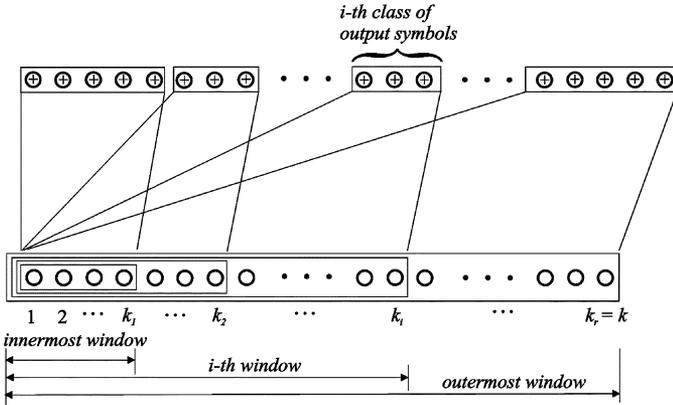


Fig. 1. Expanding window fountain (EWF) codes.

codes are EWF codes defined by a single window, i.e., all the input symbols are of equal importance.

In [16] and [17], the asymptotic probabilities (as the source block length  $k \rightarrow \infty$ ) of input symbols belonging to different importance classes of EWF codes, after  $l$  decoding iterations of the iterative BP decoder [24], are derived. Evolution of the erasure probabilities with the iterations of the BP decoding algorithm are obtained using the generalized and-or-tree analysis which we summarize in the following lemma (cf., [16] and [17]).

**Lemma 2.1:** For an EWF code  $\mathcal{F}_{EW}(\Pi, \Gamma, \Omega^{(1)}, \dots, \Omega^{(r)})$ , the probability  $y_{l,j}$  that the input symbol of class  $j$  is not recovered at the receiver upon collecting  $(1 + \epsilon)k$  encoded symbols, where  $\epsilon$  is the reception overhead, after  $l$  iterations of the iterative BP decoder is shown in (1) at the bottom of the page.

For a given EWF code, this lemma outputs the set of asymptotic recovery probabilities for input symbols of different importance classes. We will use these probabilities in Sections III and IV, when optimizing EWF codes for scalable video source output with respect to selected end-to-end video distortion measures.

In the following, we present a design example for the simple case of EWF code with two importance classes. The expressions for the erasure probabilities of most important bit (MIB) class and least important bit (LIB) class after  $l$  iterations,  $y_{l,1}$  and  $y_{l,2}$ , for an EWF code  $\mathcal{F}_{EW}(\Pi_1 x + \Pi_2 x^2, \Gamma_1 x + \Gamma_2 x^2, \Omega^{(1)}, \Omega^{(2)})$ , with the reception overhead fixed to  $\epsilon$ , can be easily obtained from the previous lemma (see [16] and [17]). For the numerical example, let the code under consideration be the EWF code  $\mathcal{F}_{EW}(\Pi(x) = 0.1x + 0.9x^2, \Gamma(x) = \Gamma_1 x + (1 - \Gamma_1)x^2, \Omega^R(x), \Omega^R(x))$  and the reception overhead equal to  $\epsilon = 0.05$ . In other words, one tenth of the source block is considered more important, and the degree distributions applied on both

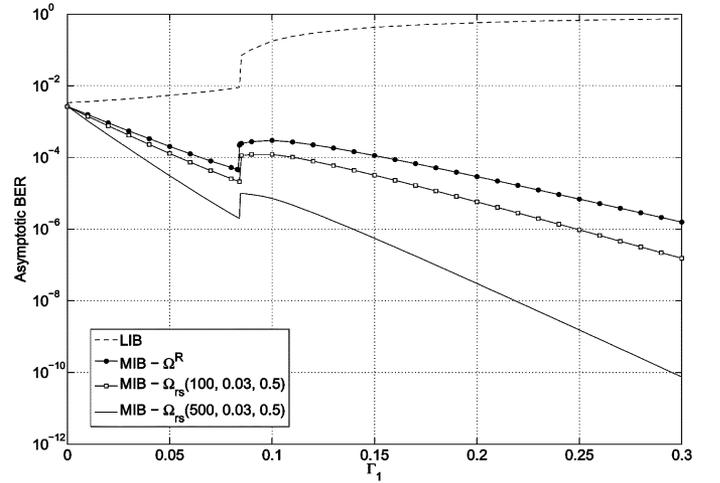


Fig. 2. Asymptotic analysis of MIB and LIB BER performance versus  $\Gamma_1$  for  $\mathcal{F}_{EW}(\Pi(x) = 0.1x + 0.9x^2, \Gamma(x) = \Gamma_1 x + (1 - \Gamma_1)x^2, \Omega^{(1)}, \Omega^{(2)})$  EWF code and reception overhead  $\epsilon = 0.05$ .

windows are the same and equal to the ‘‘Raptor’’ degree distribution [7]:

$$\begin{aligned} \Omega^R(x) = & 0.007969x + 0.493570x^2 \\ & + 0.166220x^3 + 0.072646x^4 + 0.082558x^5 \\ & + 0.056058x^8 + 0.037229x^9 + 0.055590x^{19} \\ & + 0.025023x^{64} + 0.003135x^{66}. \end{aligned} \quad (2)$$

Asymptotic erasure probabilities (assuming  $l \rightarrow \infty$ ) for MIB class,  $y_{\infty,1}$ , and LIB class,  $y_{\infty,2}$ , as a function of the first window selection probability  $\Gamma_1$ , are presented in Fig. 2. The figure demonstrates significant bit error rate (BER) improvement of MIB class over the BER of standard EEP LT codes (that corresponds to the point  $\Gamma_1 = 0$ ), with negligible loss in LIB class BER. The trend of MIB BER improvement continues until the ‘‘threshold’’ value of the first window selection probability  $\Gamma_1$ , which equals  $\Gamma_1 = 0.084$  in this example, upon which LIB BER deteriorates significantly. Also, Fig. 2 presents an important feature of EWF codes: MIB BER performance of EWF codes improves by enhancing the degree distribution  $\Omega^{(1)}(x)$  applied on the MIB window. In our example, when instead of the Raptor distribution  $\Omega^R(x)$ , we apply ‘‘stronger’’ but computationally more expensive truncated robust soliton distributions  $\Omega_{rs}(k_{rs}, \delta, c)$  [16], [17] with the maximum degrees bounded to  $k_{rs} = 100$  and  $k_{rs} = 500$ , we obtain significant MIB BER performance improvements.

In Fig. 3, for EWF code  $\mathcal{F}_{EW}(\Pi(x) = 0.1x + 0.9x^2, \Gamma(x) = 0.084x + 0.916x^2, \Omega_{rs}(500, 0.5, 0.03), \Omega^R(x))$ , we can track the asymptotic

$$\begin{aligned} y_{0,j} &= 1 \\ y_{l,j} &= e^{-(1+\epsilon) \sum_{i=j}^r (\Gamma_i / \sum_{t=1}^i \Pi_t) \Omega^{(i)'}} \left( 1 - \left( \sum_{m=1}^i \Pi_m y_{l-1,m} / \sum_{t=1}^i \Pi_t \right) \right) \end{aligned} \quad (1)$$

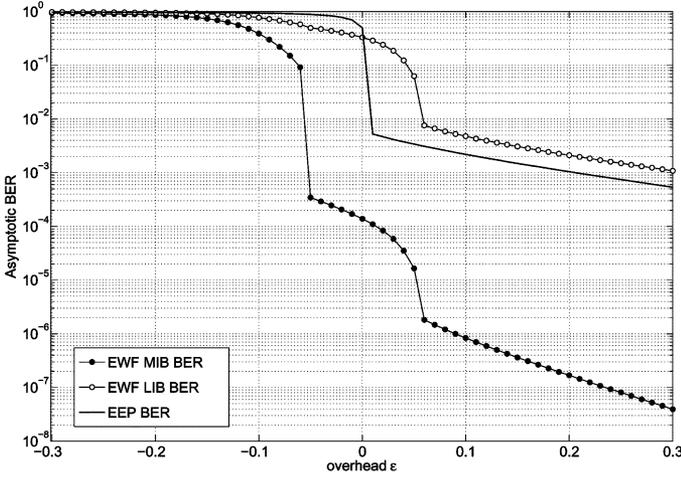


Fig. 3. Asymptotic analysis of MIB and LIB BER performance versus overhead  $\epsilon$  for  $\mathcal{F}_{EW}$  ( $\Pi(x) = 0.1x + 0.9x^2$ ,  $\Gamma(x) = 0.084x + 0.916x^2$ ,  $\Omega^{(1)}, \Omega^{(2)}$ ) EWF code.

erasure probabilities of MIB and LIB classes of source symbols as a function of overhead  $\epsilon$  of encoded data collected at the receiver. The figure clearly shows that, as compared to the standard EEP LT codes, EWF codes enable earlier and more reliable recovery of MIB class, for the price of small deterioration of LIB class recovery performance.

### III. SCALABLE VIDEO MULTICAST USING EWF CODES

Digital fountain transmission over lossy packet networks is universally capacity approaching for erasure channel associated with any receiver, given that potentially infinite amount of encoded symbols can be created at the encoder and sent to the receivers. For real-time scalable video streaming, fountain multicast solutions are usually concerned with two problems. First, the amount of the encoded symbols sent is finite; moreover, for many delay-constrained applications, the amount of encoded data received by a receiver per source block might be severely limited (e.g., due to low receiver access bandwidth, or poor channel conditions). With slightly less amount of received encoded symbols than needed for successful decoding, the iterative decoder is typically able to reconstruct only a negligible part of the data block transmitted, due to the typical “avalanche decoding” behavior. Second, standard fountain codes are EEP codes, whereas scalable video transmission calls for UEP FEC schemes due to the unequal importance of data in the source blocks of scalable bitstream. In other words, when scalable video, e.g., H.264 SVC, is coupled with existing FEC standards (RS, LT, Raptor), the output bitstream loses scalable property, resulting in non-efficient video distribution.

In fountain-based multicast systems proposed in [22], data multicast transmission proceeds in two phases. In the first phase, enough encoded packets are sent to facilitate successful decoding for most of the receivers. If some receivers cannot collect enough encoded data to finish decoding, by feedback signaling, they indicate their participation in the second, repair phase, where only this subset of receivers is fed by a new stream of encoded data. This scheme is suitable for applications that are not delay constrained. In both transmission phases,

the same fountain code is used and all the source data are given the same priority, regardless of the source characteristics, receiver bandwidths and channel conditions. Alternatively, in [8], [9], and [12], the schemes with independent codes associated with different enhancement layers of a video stream are proposed. Multiple fountain sources are then allocated with different transmission rates providing the overall UEP behavior. However, using multiple fountain codes requires rather complex rate allocation optimization and control of the encoding process. Also, unlike EWF codes which are applied over the whole source block, separate rateless codes applied over each enhancement layer are shorter (as each one is applied over a part of the source block) and therefore less efficient.

In the following, we describe a scalable EWF-based video multicast scheme that addresses the aforementioned problems using a single-code solution at the transmitter side. The EWF solution adapts the real-time scalable video stream delay constraints and unequal data importance to the reception conditions of heterogeneous receiver classes. Due to UEP and unequal recovery time (URT) properties of EWF codes, reconstruction of the more important parts of the source block is more reliable and happens earlier.

#### A. Scalable EWF Multicast: System Setting

We consider a scenario where a real-time scalable coded video stream is transmitted from a video server to a number of heterogeneous receivers over a lossy packet networks, as illustrated in Fig. 4. At the video server side, the scalable coded video stream is periodically broken into the source blocks, and each source block is separately encoded by an EWF encoder. We assume that each source block consists of an equal number of  $k$  symbols, and that the importance of data decreases from the beginning towards the end of the block. Typically, each source block contains one group of frames (GOF) of the scalable video information stream. Due to real-time constraints, the video server is able to produce “only” a finite amount of  $(1 + \epsilon_S)k$  EWF encoded symbols before moving on to the next source block. The source overhead,  $\epsilon_S > 0$ , is determined by the video server capabilities and/or the bandwidth of the access link. We assume a setting with a single EWF video streaming server, although by the same argument as for the standard fountain codes, the system can be easily implemented with multiple EWF video streaming servers.

EWF encoded symbols are transmitted in a multicast session to heterogeneous receivers. We classify receivers into  $r$  receiver classes based on their reception capabilities and channel quality. The  $i$ th receiver class,  $1 \leq i \leq r$ , is defined by the reception overhead  $\epsilon_{R,i}$ , where  $\epsilon_{R,i} \leq \epsilon_S$ , i.e., the receiver in the  $i$ th class is able to collect  $(1 + \epsilon_{R,i})k$  EWF encoded symbols for each source block, out of the  $(1 + \epsilon_S)k$  symbols transmitted. We assume that  $\epsilon_{R,i} < \epsilon_{R,j}$  if  $i < j$ , i.e., the receiver capabilities increase with the receiver class index  $i$ .

The task of the EWF encoder is to supply the receivers with scalable video source blocks through EWF encoded data streams. Due to different reception capabilities of different receiver classes, our goal is to match the EWF encoded data stream to each receiver class simultaneously, that is, the first receiver class (with the worst reception conditions) should be

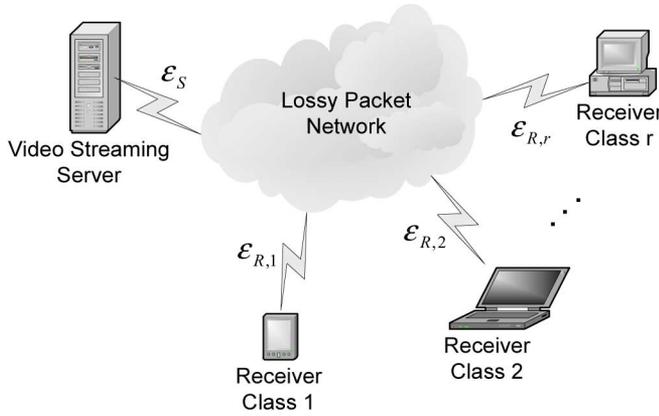


Fig. 4. Scalable video multicast to heterogeneous receiver classes.

able to recover the first part (the most important part) of the source block with high probability, the second receiver class should be able to recover the first two parts of the source block with high probability, etc. To do so, we design EWF code with  $r$  expanding windows by optimizing the set of design parameters  $\{\Pi(x), \Gamma(x), \Omega^{(1)}(x), \dots, \Omega^{(r)}(x)\}$ . EWF encoding with finite source overhead  $\epsilon_S$  is then applied at the source, across each source block to be transmitted, as described in Section II.

### B. Scalable EWF Multicast: System Design

With the described setting, the scalable EWF multicast system design reduces to the design of the EWF code such that given quality-of-service (QoS) guarantees for different receiver classes are satisfied. In this section, we discuss a general approach where, as QoS parameters, we select the probabilities of complete reconstruction of different importance classes of the source block at the receivers in different receiver classes. In the later sections, the EWF code optimization is further extended using more specific video distortion measures.

For a given reception overhead of the receivers belonging to the  $i$ th receiver class,  $\epsilon_{R,i}$ , and the parameters of selected EWF code  $\mathcal{F}_{EW}(\Pi, \Gamma, \Omega^{(1)}, \dots, \Omega^{(r)})$ , we can calculate (asymptotic) erasure probabilities of input symbols in each of the  $r$  importance classes. Let  $p_i^{(j)}$  denote erasure probability of the input symbol of the  $i$ th importance class at the  $j$ th receiver class. Using  $p_i^{(j)}$ , and under the asymptotic assumption that the probabilities  $p_i^{(j)}$  for different input symbols of the same class are independent, we can calculate the probability  $P_i^{(j)}$  that the  $i$ th importance class of the source block is completely recovered by the  $j$ th receiver class:

$$P_i^{(j)} = \left(1 - p_i^{(j)}\right)^{s_i} \quad (3)$$

where  $s_i$  is the number of input symbols in the  $i$ th importance class of the source block.

We will use the set of probabilities  $P_i^{(j)}$  to define QoS guarantees for each receiver class of the proposed scalable EWF multicast system. It is worth noting that  $P_i^{(j)} < P_i^{(k)}$  for  $j < k$  due to Lemma 2.1 and  $\epsilon_{R,j} < \epsilon_{R,k}$ ; that is, a receiver in the better class will be able to satisfy all the QoS guarantees imposed on the receiver in the worse class. Therefore, it is convenient to define

QoS guarantees for the scalable EWF multicast system as the following set of probabilities:  $\{P_1^{(1)}, P_2^{(2)}, \dots, P_r^{(r)}\}$ . In other words, for the  $i$ th receiver class, we define only QoS guarantee  $P_i^{(i)}$  for reconstruction of the input symbols of the  $i$ th class. QoS guarantees for more important classes of input symbols are already implicitly included in the QoS guarantees  $P_j^{(j)}$  of the receiver classes indexed with  $j < i$ . For input symbols that belong to classes  $j > i$ , which are of less importance, the  $i$ th receiver class is not provided with any QoS guarantees.

Before proceeding further, we shortly summarize the EWF code design problem for scalable EWF multicast system. Our goal is to find the set of EWF code design parameters  $\{\Pi, \Gamma, \Omega^{(1)}, \dots, \Omega^{(r)}\}$  such that the corresponding EWF code satisfies the performance threshold  $\mathbf{P}_{th} = (P_1^{(1)}, P_2^{(2)}, \dots, P_r^{(r)})$  for the different receiver classes, given their reception capabilities  $\epsilon_{\mathbf{R}} = (\epsilon_{R,1}, \epsilon_{R,2}, \dots, \epsilon_{R,r})$ . More detailed exploration on the possible EWF code design scenarios for scalable EWF multicast can be found in [23].

### C. Scalable EWF Multicast: Numerical Example

In this subsection, we provide a numerical example for the design of EWF code  $\mathcal{F}_{EW}(\Pi, \Gamma, \Omega^{(1)}, \dots, \Omega^{(r)})$  that meets the requested QoS guarantees  $\mathbf{P}_{th}$  for a given reception performance  $\epsilon_{\mathbf{R}}$ . For simplicity, we assume a setting with  $r = 2$  receiver classes (i.e., EWF code with two expanding windows), the MIB and the LIB class. We simplify the EWF code design by setting the distribution on the first window as “stronger” truncated robust soliton distribution  $\Omega^{(1)}(x) = \Omega_{r,s}(k_{r,s}, 0.5, 0.03)$ , where  $k_{r,s} = k_1 = \Pi_1 k$ , and on the second window the “weaker” constant average Raptor degree distribution  $\Omega^{(2)}(x) = \Omega_{r,s}(\Pi_1 k, 0.5, 0.03)$ . With these simplifications, the design of the EWF code  $\mathcal{F}_{EW}(\Pi_1 x + (1 - \Pi_1)x^2, \Gamma_1 x + (1 - \Gamma_1)x^2, \Omega_{r,s}(\Pi_1 k, 0.5, 0.03), \Omega^{(2)}(x))$  is determined by two independent variables:  $\Pi_1$  and  $\Gamma_1$  (the first window selection probability and the fraction of the data contained in it). In general, as a result of this design process, we obtain a set (region) of possible  $(\Pi_1, \Gamma_1)$  pairs that satisfy required QoS conditions. Note that, depending on the values of  $\epsilon_{\mathbf{R}}$  and  $\mathbf{P}_{th}$ , this set can be empty, providing no solution for the requested scenario.

As an example, we select the following constraints:  $\epsilon_{\mathbf{R}} = (0.1, 1)$  and  $\mathbf{P}_{th} = (0.95, 0.8)$  and the source block length of  $k = 3800$  symbols. In other words, we have two classes of receivers: the first, worse class, characterized by the 10% reception overhead, and the second, better class, with 100% overhead. The QoS guarantees require that a receiver in the worse class has a probability of reconstruction of the MIB block of at least 95%, while a receiver in the better class should, in addition, be able to reconstruct the LIB block with probability of at least 80%. The reconstruction probabilities of the MIB block for the worse class of receivers,  $P_1^{(1)}$ , and the LIB block for the better class of receivers,  $P_2^{(2)}$ , are given as functions of two variables  $(\Pi_1, \Gamma_1)$  in Fig. 5(a) and (b), respectively. In both figures, one can track changes of probabilities  $P_1^{(1)}$  and  $P_2^{(2)}$  which are illustratively represented by differently shaded gray regions. The solution region of  $(\Pi_1, \Gamma_1)$  pairs that satisfy given constraints  $\epsilon_{\mathbf{R}} = (0.1, 1)$  and  $\mathbf{P}_{th} = (0.95, 0.8)$  is presented in Fig. 5(c).

From the solution of the given design scenario, which is a set of  $(\Pi_1, \Gamma_1)$  pairs, we select operational pair  $(\Pi_1, \Gamma_1)$  using a suitable criterion. One way to proceed would be to select a solution that maximizes the  $\Pi_1$  value, i.e., to place as much as possible data into the more important class. In this example, such a solution is the point  $(\Pi_1, \Gamma_1) = (0.365, 0.205)$  that treats 36.5% of the transmitted source block as the more important data. However, other optimality considerations of points in the  $(\Pi_1, \Gamma_1)$  region are possible, particularly in the case when the information source is a scalable video coder. We provide some examples in Section IV.

#### IV. DISTORTION-OPTIMAL SCALABLE EWF VIDEO MULTICAST

The design methodology described in the previous section can be applied to any kind of data. As a result, the set of  $(\Pi_1, \Gamma_1)$  pairs is obtained that provide QoS performance guarantees for each class of receivers in terms of the reconstruction probabilities of different importance classes. For a specific case of a scalable video data, different points from the  $(\Pi_1, \Gamma_1)$  region will have different performance in terms of the quality of the reconstructed video.

Scalable video coders are particularly useful in multicast scenarios, due to the fact that they efficiently accommodate receivers with different data rates and/or channel conditions. The output bitstream of a scalable video encoder is segmented into layers of progressively decreasing importance, so that receivers with better reception conditions that receive more layers will obtain a higher video quality. This makes a scalable video coder together with a scalable EWF multicast system a promising combination for multicast multimedia distribution services. Since EWF codes are a UEP fountain solution flexible to design, they can be easily adapted to a multi-layer scalable coded bitstream offering more protection to more important layers. Optimizing EWF codes may be performed jointly with error-resilience mechanisms at lower layers, providing a powerful cross-layer optimized multicast fountain solution.

To select the “optimal”  $(\Pi_1, \Gamma_1)$  point from the region obtained as an output of the EWF multicast system design, we apply the distortion-based optimization that takes into account the expected video distortion at the receiving end [1]. This performance criterion is based on the fact that the reconstruction process at a scalable video decoder deteriorates significantly after the first transmission error is encountered, due to an error propagation effect. Therefore, we adopt the strategy to stop decoding after detecting the first uncorrected symbol error, and thus use only error-free layers for reconstruction, though different error concealment tools can be applied to enable decoding after a symbol error occurs. However, in this paper, we focus exclusively on error-correcting capabilities of EWF codes and do not consider any error concealment tools.

We assume that the source block is divided into  $r$  layers of lengths  $s_1, s_2, \dots, s_r$  symbols. The importance of data contained in layers decreases from the first towards the last layer in the block. Reconstruction of the source block at the receiver is based on correctly received consecutive layers until the first

layer for which a channel transmission error is detected. We denote a probability of correct reconstruction of each of  $r$  data layers as  $P_1, P_2, \dots, P_r$ .

The transmission scheme that minimizes the expected distortion of the video reconstructed at the receiver is considered to be distortion-based optimal. In case of the expected peak signal-to-noise ratio (PSNR) measure, it is equal to

$$PSNR_{avg} = \sum_{i=0}^r P(i) \cdot PSNR(i) \quad (4)$$

where  $P(i)$  is the probability that the first  $i$  consecutive layers are correctly received:

$$P(i) = \begin{cases} 1 - P_1, & \text{for } i = 0 \\ \prod_{j=1}^i P_j \cdot (1 - P_{i+1}), & \text{for } i = 1, 2, \dots, r - 1 \\ \prod_{j=1}^r P_j, & \text{for } i = r \end{cases} \quad (5)$$

where  $PSNR(0) = 0$ , and for  $i > 0$ ,  $PSNR(i)$  is the PSNR upon the complete recovery of  $i$  layers, averaged over all frames of the video segment. Note that  $PSNR(i)$  can be either calculated offline, or different models can be used (see [1]). It is worth noting that the average PSNR is a function of both the type of a scalable video coder used and the content of the data transmitted.

In the multicast scenario, where we are dealing with a number of receiver classes,  $PSNR_{avg}$  have to be averaged over all the classes:

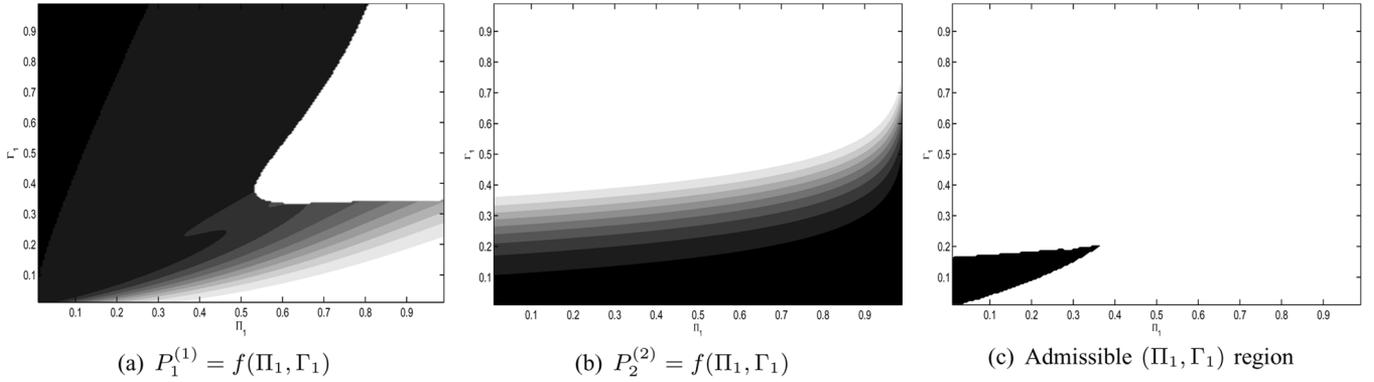
$$PSNR_{avg} = \frac{1}{r} \sum_{j=1}^r PSNR_{avg}^{(j)} \quad (6)$$

where  $PSNR_{avg}^{(j)}$  is the average PSNR at the receiver of the  $j$ th receiver class.

##### A. H.264 SVC EWF Multicast: Numerical Optimization

In this subsection, we present an example of the distortion-based optimized EWF code design where the video server is multicasting H.264 SVC coded video stream. H.264 SVC [19], standardized recently by the Joint Video Team of the ITU-T VCEG and the ISO/IEC MPEG, is the scalable extension of H.264/AVC [18]. H.264 SVC outperforms previous scalable video coders while providing temporal, spatial, and quality scalability with backwards compatibility with H.264/AVC. It maintains key features of H.264/AVC while introducing new tools necessary for maximizing scalability, such as new inter-layer prediction of motion and residual, the concept of key pictures, single motion compensation loop decoding providing a decoder complexity close to that of single-layer coding.

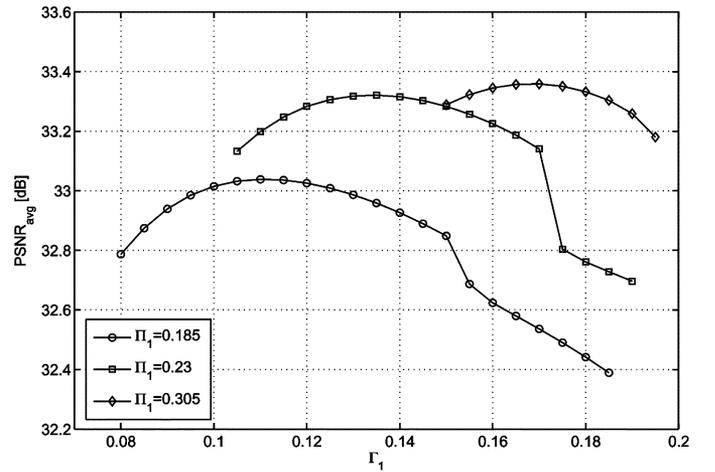
We use the CIF *Stefan* video sequence (30 fps,  $352 \times 288$ ) with the base layer (BL) and fourteen enhancement layers (EL) which gradually improve the overall video quality. The video sequence is segmented into GOFs of size 16 frames, and every 16/30 s, the EWF encoder is supplied by a new GOF data as the source block. The source block size is approximately 190 000 bytes and, assuming symbol size of 50 bytes, we obtain

Fig. 5. Region of  $(\Pi_1, \Gamma_1)$  that satisfies given constraints  $\epsilon_R$  and  $P_{th}$ .TABLE I  
MIB WINDOW CONTENT FOR H.264 SVC STEFAN SEQUENCE

MIB Window Content	$k_1$	$\Pi_1$	Bit Rate [kbps]	Y-PSNR [dB]
BL only	400	0.105	292.37	25.79
BL + 1 EL	700	0.185	510.65	27.25
BL + 2 ELs	875	0.23	636.56	28.14
BL + 3 ELs	1155	0.305	839.82	29
BL + 4 ELs	1550	0.41	1127.1	29.51
BL + All ELs	3800	1	2764.55	40.28

the source block size of  $k = 3800$  symbols (as analyzed in Section III-B). We assume that the base layer is always placed in the first window, with the minimum window size necessary to accommodate the base layer data set to  $k_1 = 400$  symbols. Apart from the base layer, we may place additional enhancement layers together with the base layer inside the first window. Several possible divisions into MIB and LIB data blocks are presented for a single (first) GOP of the video sequence in Table I, where for each division, the corresponding values of the first window absolute and relative size ( $k_1$  and  $\Pi_1$ ), MIB data bit rate, and the average PSNR of the Y component upon complete recovery of the first window are presented.

We apply the distortion-based optimization on the system setting presented in Section III-C in order to find the optimal  $(\Pi_1, \Gamma_1)$  point in the region given in Fig. 5(c). However, as the division of the source block into MIB and LIB class is dictated by the number of ELs placed in the first window, we perform the distortion-based optimization only for the values of  $\Pi_1$  that accommodate BL and whole number of ELs (for example, the values of  $\Pi_1$  in Table I). Therefore, our optimization problem reduces to the optimization of  $\Gamma_1$  for a fixed value of  $\Pi_1$ , for the interval of  $\Gamma_1$  values such that the pairs  $(\Pi_1, \Gamma_1)$  are inside the region in Fig. 5(c). For each pair  $(\Pi_1, \Gamma_1)$  and for each of the two receiver classes defined by their reception capabilities  $\epsilon_R = (0.1, 1)$ , we can calculate the probabilities  $P_1^{(j)}$  and  $P_2^{(j)}$  of complete recovery of the MIB and LIB blocks at the  $j$ th receiver class. Replacing these probabilities as the layer reconstruction probabilities  $P_1$  and  $P_2$  in (5), and using (4) with the appropriate values of PSNRs from Table I, we can calculate the

Fig. 6. Numerical example of  $\Gamma_1$  optimization for the values of  $\Pi_1 = \{0.185, 0.23, 0.305\}$ .

average PSNR values,  $PSNR_{avg}^{(1)}$  and  $PSNR_{avg}^{(2)}$ , for both receiver classes.

Fig. 6 provides an example of  $\Gamma_1$  optimization for the following values of  $\Pi_1 = \{0.185, 0.23, 0.305\}$ , i.e., when the first window contains BL and one, two or three ELs, respectively (Table I). For each value of  $\Pi_1$ , we optimize  $\Gamma_1$  over the range of values such that the pairs  $(\Pi_1, \Gamma_1)$  satisfy QoS constraint  $P_{th} = (0.95, 0.8)$  [Fig. 5(c)]. The optimum values  $\Gamma_1 = \{0.11, 0.135, 0.17\}$  are found that provide maximum average PSNR values of  $PSNR_{avg} = \{33.036, 33.321, 33.359\}$  dB for each MIB window size  $\Pi_1 = \{0.185, 0.23, 0.305\}$ , respectively. Average PSNR values for worse and better receiver class are  $PSNR_{avg}^{(1)} = \{26.982, 27.676, 28.07\}$  and  $PSNR_{avg}^{(2)} = \{39.09, 38.966, 38.647\}$ , respectively. These values demonstrate the ability of EWF codes to simultaneously support heterogeneous receiver classes, matching the received video quality with the reception capability of the receiver class.

### B. H.264 SVC EWF Multicast: Simulation Results

The numerical optimization results presented in Sections III-C and IV-A are derived using asymptotic erasure probabilities of input symbols obtained from analytical expressions presented in Section II. These analytical expressions

assume infinite source block length ( $k \rightarrow \infty$ ), whereas in the design example, we deal with the finite-length EWF system scenario ( $k = 3800$ ). To verify that numerical results are a good approximation of the “real-world” behavior, we perform simulation experiments. We select a simulation setting identical to the numerical optimization setting presented in Sections III-C and IV-A. We assume two scenarios, both with two receiver classes, where the first one is described by receiver reception capabilities  $\epsilon_R^{(A)} = (0.1, 1)$ , and the second one is described by receiver reception capabilities  $\epsilon_R^{(B)} = (0.35, 1.5)$ . We refer to these scenarios as scenario A and scenario B, respectively.<sup>1</sup>

Firstly, we provide the simulation results that correspond to the setting described in Section III-C. In other words, we do not specify the information source, but only request that given QoS constraints  $P_{th} = (0.95, 0.8)$  are met at the receivers of both receiver classes. Our goal is to find the region of pairs  $(\Pi_1, \Gamma_1)$  that satisfy QoS constraints  $P_{th}$ . As the simulation experiments are considerably more time consuming than the numerical optimization, we simulate the source block transmission only for the finite set  $\Pi_1 = \{0.185, 0.23, 0.305, 0.405\}$  of  $\Pi_1$  values and for each value, we search for the interval of  $\Gamma_1$  values that satisfy  $P_{th}$ . In each simulation run [i.e., for a fixed pair of values  $(\Pi_1, \Gamma_1)$ ], the total number of the source blocks transmitted is set to 3000. Out of the total number of transmitted source blocks, we determine the number of unsuccessfully decoded blocks (both MIB and LIB blocks), the number of blocks with successfully decoded MIB block, and the number of blocks with complete recovery of both MIB and LIB block. These measurements are used to determine the corresponding  $P_1^{(1)}$  and  $P_2^{(2)}$  values.

Fig. 7 presents the simulation results for the case  $\Pi_1 = 0.230$  and the receiver classes of scenario A. The interval of  $\Gamma_1$  values satisfying the QoS constraints  $P_{th} = (0.95, 0.8)$  is the region approximately located between  $\Gamma_{1,\min} = 0.093$  and  $\Gamma_{1,\max} = 0.192$  (as determined by vertical dash-dot lines in Fig. 7). Similar procedures are repeated for all the values from the set  $\Pi_1 = \{0.185, 0.23, 0.305, 0.405\}$ . The results, i.e., the corresponding solution intervals  $[\Gamma_{1,\min}, \Gamma_{1,\max}]$ , are tabulated in Table II.

The simulation results from Table II are presented in Fig. 8(a) and compared with the results obtained numerically in Section III-C [Fig. 8(b), which is a zoomed in version of Fig. 5(c)]. Fig. 8(c) and (d) presents the numerical solution and simulation results for scenario B ( $\epsilon_R^{(B)} = (0.35, 1.5)$ ), respectively. In both scenarios, there is an excellent match between the results predicted by theory (Section III-C) and the results obtained using simulation experiments. This demonstrates that, although the simulation results are applied on the finite-length scenario, the EWF codelengths of several thousands symbols are already sufficient to confirm the analysis based on the asymptotic probability expressions.

Secondly, we provide simulation results to confirm the distortion-based optimized EWF code design example, as de-

<sup>1</sup>In both scenarios, all the receiver classes are defined with positive reception overheads  $\epsilon_R$ . However, due to URT property of EWF codes, optimization is possible even for negative  $\epsilon_R$  values. An example of this scenario is available in [25].

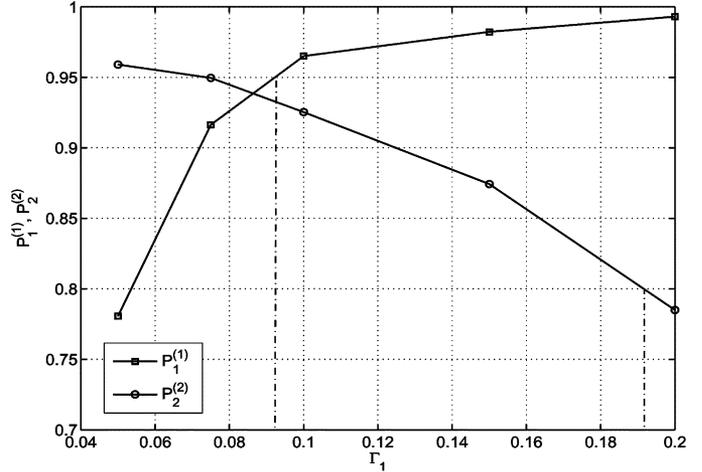


Fig. 7. Set of  $\Gamma_1$  values satisfying  $P_{th} = (0.95, 0.8)$  for  $\Pi_1 = 0.23$  and scenario A.

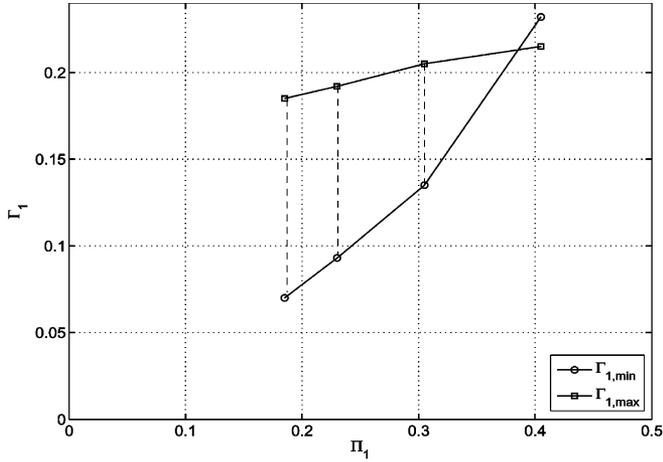
TABLE II  
SOLUTION INTERVALS OF  $\Gamma_1$  FOR DIFFERENT  $\Pi_1$

MIB Window Size $\Pi_1$	Scenario A $[\Gamma_{1,\min}, \Gamma_{1,\max}]$	Scenario B $[\Gamma_{1,\min}, \Gamma_{1,\max}]$
0.185	[0.070, 0.185]	[0.027, 0.345]
0.230	[0.093, 0.192]	[0.037, 0.350]
0.305	[0.135, 0.205]	[0.055, 0.365]
0.405	[0.232, 0.215]	[0.087, 0.370]

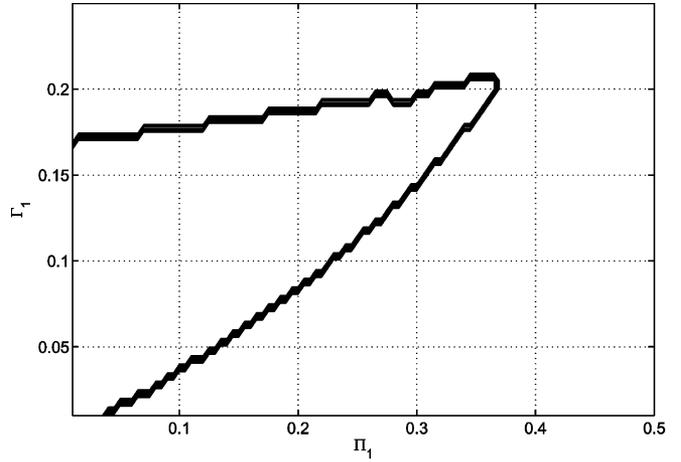
scribed in Section IV-A, where the server is multicasting H.264 SVC coded video stream. We keep the same assumptions on the transmitted CIF *Stefan* video sequence (Table I) and EWF encoding process as for the numerical optimization. The set of possible MIB window lengths analyzed in simulations is  $\Pi_1 = \{0.185, 0.23, 0.305\}$ . To calculate average PSNR value for each  $\Pi_1$ , similar experiments are performed using the same set of  $\Gamma_1$  values. The results obtained using (4) and (5) for  $\Pi_1 = 0.23$  and scenario A are presented in Fig. 9 where they are compared with the results obtained by numerical optimization in Section IV-A. It can be seen that simulation results correspond to the results obtained by numerical optimization. Similar procedures are repeated for other  $\Pi_1$  values which are shown to produce a very good match with the numerical results.

### C. H.264 SVC EWF Multicast: Precoded EWF Codes

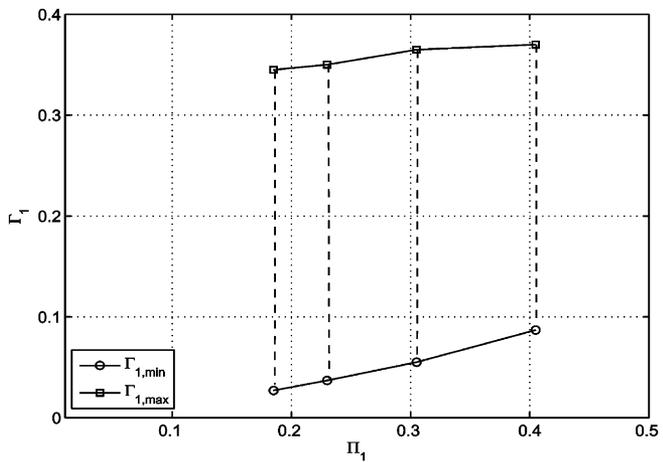
In this section, we demonstrate how results presented in this paper can be additionally improved by concatenating EWF code to a high-rate LDPC precode, in a similar way to improvement that Raptor codes offer over LT codes [7]. We have seen that the linear encoding/decoding complexity of an EWF code is achieved by applying the constant average degree distributions over its windows. However, this results in an increase of error floors in the asymptotic recovery probabilities, especially for the low importance symbols. Higher error floors combined with the assumption (3) that each input symbol of the  $i$ th importance



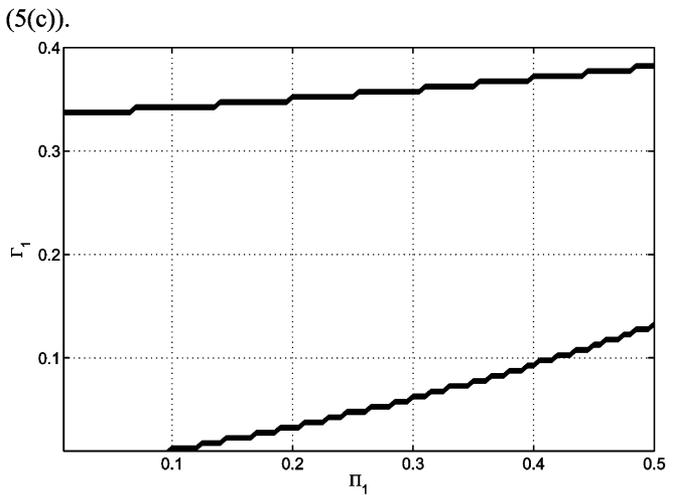
(a) Admissible  $(\Pi_1, \Gamma_1)$  region: scenario A, simulation results.



(b) Admissible  $(\Pi_1, \Gamma_1)$  region: scenario A, numerical results



(c) Admissible  $(\Pi_1, \Gamma_1)$  region: scenario B, simulation results.



(d) Admissible  $(\Pi_1, \Gamma_1)$  region: scenario B, numerical results.

Fig. 8. Admissible  $(\Pi_1, \Gamma_1)$  region: comparison of numerical and simulation results.

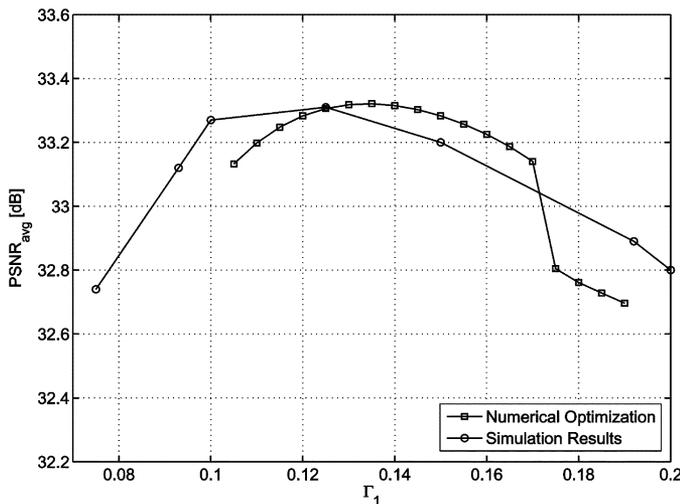


Fig. 9. Numerical optimization versus simulation results of  $PSNR_{avg}$  values for  $\Pi_1 = 0.23$  and scenario A.

class at the  $j$ th receiver class is decoded independently with probability  $p_i^{(j)}$  gives modest values of the probability  $P_i^{(j)}$  of

the complete reconstruction of the  $i$ th importance class source block at the  $j$ th receiver class. Thus, even at the overheads as high as  $\epsilon_{R,2} = 1$  in the example setting explored in the previous subsections, the QoS constraint of  $P_2^{(2)} = 0.8$  is obtained. In order to increase QoS guarantees, we add redundancy within each of the importance class prior to the EWF encoding process by precoding each of the importance class source blocks by a high-rate LDPC code (Fig. 10). This way, once the decoding of an EWF code allows recovery of a sufficient fraction of the importance class source block, the LDPC precode should be able to “finish off” decoding with a vanishing probability of error, resulting in dramatically increased probability  $P_i^{(j)}$  of complete importance class reconstruction.

We performed simulation experiments to determine the performance of a precoded EWF code with degree distribution  $\Omega^R(x)$  over both windows and the source block of length  $k = 3800$  symbols. Each of the importance class source blocks was encoded by a systematic hybrid LDPC-Half code [14]. The admissible  $(\Pi_1, \Gamma_1)$  region for significantly restricted reception capabilities  $\epsilon_R = (0.05, 0.2)$ , and at the same time more demanding QoS guarantees  $\mathbf{P}_{th} = (0.99, 0.95)$ , as compared to the previous sections, is presented in Fig. 11. We note that

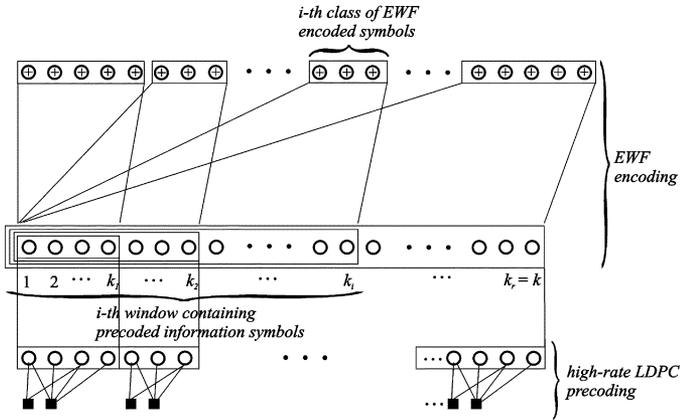
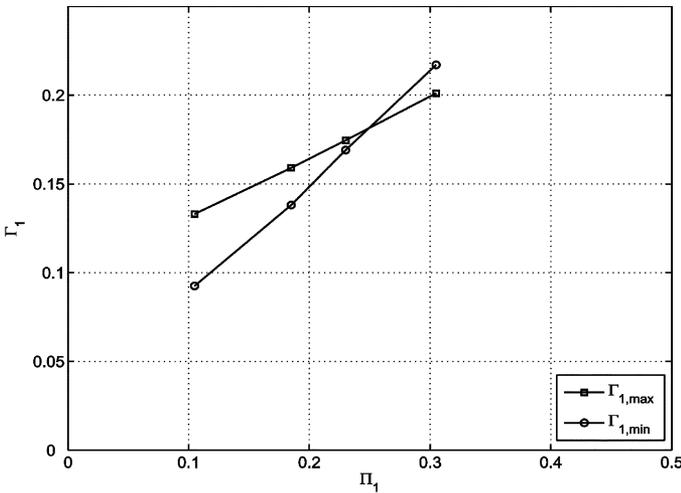


Fig. 10. Precoded EWF codes.

Fig. 11. Admissible  $(\Pi_1, \Gamma_1)$  region for precoded EWF codes.

this preliminary investigation does indicate that additional improvement of the robust scalable H.264 multicast is possible by combining low block-error-rates achieved by precoding with the flexibility of the EWF design.

## V. CONCLUSIONS

A novel scalable multicast system based on EWF codes is proposed as an efficient and flexible solution for real-time multimedia delivery to various classes of receivers with different reception conditions. In this scenario, classical EEP fountain codes would perform poorly due to a potentially large number of receivers not being able to collect enough encoded symbols to perform successful decoding. EWF code design is shown to offer simple “single-code” solution, where the applied EWF code can be optimized to adapt the encoded data stream to satisfy QoS guarantees offered to each receiver class. We demonstrated that, in the case of a scalable video source, further distortion-based optimization of EWF codes is possible for improved video reception quality. A detailed analysis of EWF multicast system design for H.264 SVC video streaming is presented both using analytical tools and simulation experiments. Numerical examples of EWF code performance obtained using analytical tools match extremely well with the corresponding simulation

analysis. The obtained results illustrate promising flexibility and efficiency of EWF codes in adapting the code at the video server side to receivers with heterogeneous reception capabilities. Finally, potential improvements of EWF-based scheme by applying Raptor-like precoding is a promising direction for further investigation.

## REFERENCES

- [1] R. Hamzaoui, V. Stanković, and Z. Xiong, “Optimized error protection of scalable image bit streams,” *IEEE Signal Process. Mag.*, vol. 22, no. 6, pp. 91–107, Nov. 2005.
- [2] I. S. Reed and G. Solomon, “Polynomial codes over certain finite fields,” *SIAM J. Appl. Math.*, vol. 8, pp. 300–304, 1960.
- [3] W.-T. Tan and A. Zakhor, “Video multicast using layered FEC and scalable compression,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, no. 3, pp. 373–386, Mar. 2001.
- [4] P. A. Chou, A. E. Mohr, A. Wang, and W. Mehertra, “Error control for receiver-driven layered multicast of audio and video,” *IEEE Trans. Multimedia*, vol. 3, no. 1, pp. 108–122, Mar. 2001.
- [5] J. Byers, M. Luby, and L. Mitzenmacher, “A digital fountain approach to reliable distribution of bulk data,” *IEEE J. Select. Areas Commun.*, vol. 20, no. 8, pp. 1528–1540, Oct. 2002.
- [6] M. Luby, “LT codes,” in *Proc. 43rd Annu. IEEE Symp. Foundations of Computer Science (FOCS)*, Vancouver, BC, Canada, Nov. 2002, pp. 271–282.
- [7] A. Shokrollahi, “Raptor codes,” *IEEE Trans. Inf. Theory*, vol. 52, no. 6, pp. 2551–2567, Jun. 2006.
- [8] J. P. Wagner, J. Chakareski, and P. Frossard, “Streaming of scalable video from multiple sources using rateless codes,” in *Proc. IEEE ICME Int. Conf. Multimedia and Expo*, Toronto, ON, Canada, Jul. 2006, pp. 1501–1504.
- [9] T. Stockhammer, T. Gasiba, W. A. Samad, T. Schierl, H. Jenkac, T. Wiegand, and W. Xu, “Nested harmonic broadcasting for scalable video over mobile datacast channels,” *Wiley J.—Wireless Commun. Mobile Comput.*, vol. 7, no. 2, pp. 235–256, Feb. 2007.
- [10] Q. Xu, V. Stanković, and Z. Xiong, “Wyner-Ziv video compression and fountain codes for receiver-driven layered multicast,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 7, pp. 901–906, Jul. 2007.
- [11] T. Schierl, S. Johansen, A. Perkis, and T. Wiegand, “Rateless scalable video coding for overlay multisource streaming in MANETs,” *J. Vis. Commun. Image Represent.*, vol. 19, no. 9, pp. 500–507, Dec. 2008.
- [12] Z. Li, Y. Li, M. Chiang, and A. R. Calderbank, “Optimal transmission scheduling for scalable wireless video broadcast with rateless erasure correction code,” in *Proc. IEEE CCNC 2009*, Las Vegas, NV, Jan. 2009.
- [13] D. G. Barquero and A. Bria, “Wireless repair for broadcast transmissions in DVB-H systems,” *Wiley J.—Wireless Commun. Mobile Comput.*, vol. 9, no. 6, pp. 733–744, Jun. 2009.
- [14] Universal Mobile Telecommunications System (UMTS): Mobile Broadcast/Multicast Service (MBMS): Protocols and Codecs, 2005.
- [15] Digital Video Broadcasting (DVB): Transmission System for Handheld Terminals (DVB-H), 2004.
- [16] D. Sejdinović, D. Vukobratović, A. Doufexi, V. Šenk, and R. Piechocki, “Expanding window fountain codes for unequal error protection,” in *Proc. 41st Annu. Asilomar*, Mar. 2007.
- [17] D. Sejdinović, D. Vukobratović, A. Doufexi, Šenk, and R. Piechocki, “Expanding window fountain codes for unequal error protection,” *IEEE Trans. Commun.*, to be published.
- [18] T. Wiegand, G. J. Sullivan, G. Bjøntegaard, and A. Luthra, “Overview of the H.264/AVC video coding standard,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 560–576, Jul. 2003.
- [19] H. Schwarz, D. Marpe, and T. Wiegand, “Overview of the scalable video coding extension of the H.264/AVC standard,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 9, pp. 1103–1120, Sep. 2007.
- [20] N. Rahnavard, B. N. Vellambi, and F. Fekri, “Rateless codes with unequal error protection property,” *IEEE Trans. Inf. Theory*, vol. 53, no. 4, pp. 1521–1532, Apr. 2007.
- [21] R. G. Gallager, “Low density parity check codes,” *IEEE Trans. Inf. Theory*, vol. IT-8, no. 1, pp. 21–28, Jan. 1962.
- [22] T. Gasiba, T. Stockhammer, and W. Xu, “Reliable and efficient download delivery with Raptor codes,” in *Proc. 4th Int. Symp. Turbo Codes and Related Topics*, Munich, Germany, Apr. 2006.
- [23] D. Vukobratović, V. Stanković, D. Sejdinović, L. Stanković, and Z. Xiong, “Scalable data multicast using expanding window fountain codes,” in *Proc. 45th Annu. Allerton Conf.*, Monticello, IL, Sep. 2007.

- [24] F. R. Kschischang, B. J. Frey, and H. A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 498–519, Feb. 2001.
- [25] D. Vukobratović, V. Stanković, D. Sejdinović, L. Stanković, and Z. Xiong, "Scalable video multicast using expanding window fountain codes," in *Proc. IEEE ICME Int. Conf. Multimedia and Expo*, Hannover, Germany, Jun. 2008.



**Dejan Vukobratović** (S'02–M'08) received the Dipl.-Ing. degree in electrical engineering and the Dr.-Ing. degree in electrical engineering from the University of Novi Sad, Novi Sad, Serbia, in 2001 and 2008, respectively.

Since 2008, he has been an Assistant Professor with the Department of Power, Electronics and Communication Engineering, University of Novi Sad. From June 2009, he is on leave as Marie Curie Postdoctoral Research Fellow at the University of Strathclyde, Glasgow, U.K. His research interests

include sparse-graph codes and iterative decoding, rateless codes, and network coding.



**Vladimir Stanković** (M'04) received the M.Eng. degree from the University of Belgrade, Belgrade, Serbia, in 2000, and the Dr.-Ing. (Ph.D.) degree from the University of Leipzig, Leipzig, Germany, in 2003.

He is currently a Glasgow Research Partnership in Engineering (GRPE) Lecturer at the University of Strathclyde, Glasgow, U.K. Before joining Strathclyde, he was with Texas A&M University, College Station, as a Research Assistant Professor, and a Lecturer at Lancaster University, Lancaster, U.K.

He has published over 80 research papers in peer-reviewed scientific journals and international conference proceedings, and three book chapters in the area of distributed source and channel coding and image/video communications. His research focuses on image/video processing, network information theory, wireless communications, and wireless ad hoc/sensor networks.

Dr. Stanković serves as an Associate Editor of the *IEEE COMMUNICATIONS LETTERS*. He has been a technical program committee member of international conferences and is a member of the organizing committee of Eusipco-2009, as Tutorial Chair.



**Dino Sejdinović** (S'08) was born in 1984 in Tuzla, Bosnia-Herzegovina. He received the Dipl. Math.-Inf. degree in mathematics and theoretical computer science from the Department of Mathematics, University of Sarajevo, Sarajevo, Bosnia-Herzegovina, in 2006. He is currently pursuing the Ph.D. degree in electrical and electronic engineering at the Centre for Communications Research, University of Bristol, Bristol, U.K., as a recipient of Ph.D. studentship by Toshiba Research Europe Ltd.

His research interests include coding and information theory, graph algorithms, and wireless communications.



**Lina Stanković** (Fagoonee) (S'00–M'03) received the B.Eng. and Ph.D. degrees from Lancaster University, Lancaster, U.K., in 1999 and 2003, respectively.

She is currently a GRPE Lecturer at the University of Strathclyde, Glasgow, U.K. Before joining Strathclyde, she was at Lancaster University as a Research Associate and then a Lecturer. She has published over 50 research papers in peer-reviewed international journals, conference proceedings, and two book chapters in the area of channel coding, signal processing for wireless communications, and optical

storage. She has worked with BT Research Labs on digital video, QinetiQ on tactical DSP-enabled communications, and Philips Research Eindhoven on signal processing for optical storage. Her research interests are channel coding, DSP-enabled wireless communications, and network code design.

Dr. Stanković has been a technical program committee member of international conferences and is a member of the organizing committee of Eusipco-2009, as Tutorial Chair.



**Zixiang Xiong** (S'91–M'96–SM'02–F'07) received the Ph.D. degree in electrical engineering from the University of Illinois at Urbana-Champaign in 1996.

From 1995 to 1997, he was with Princeton University, Princeton, NJ, first as a visiting student and then as a research associate. From 1997 to 1999, he was with the University of Hawaii. Since 1999, he has been with the Department of Electrical and Computer Engineering at Texas A&M University, College Station, where he is a Professor. His research interests are network information theory, code designs and applications, biomedical engineering, genomic signal processing, and networked multimedia.

Dr. Xiong received a National Science Foundation Career Award in 1999, an Army Research Office Young Investigator Award in 2000, and an Office of Naval Research Young Investigator Award in 2001. He also received the 2006 IEEE Signal Processing Magazine best paper award. He served as an Associate Editor for the *IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY* (1999–2005), the *IEEE TRANSACTIONS ON IMAGE PROCESSING* (2002–2005), and the *IEEE TRANSACTIONS ON SIGNAL PROCESSING* (2002–2006). He is currently an Associate Editor for the *IEEE TRANSACTIONS ON COMMUNICATIONS* and the *IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS, PART B*. He was the publications chair of ICASSP'07 and the technical program committee co-chair of ITW'07.

Dr. Xiong received a National Science Foundation Career Award in 1999, an Army Research Office Young Investigator Award in 2000, and an Office of Naval Research Young Investigator Award in 2001. He also received the 2006 IEEE Signal Processing Magazine best paper award. He served as an Associate Editor for the *IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY* (1999–2005), the *IEEE TRANSACTIONS ON IMAGE PROCESSING* (2002–2005), and the *IEEE TRANSACTIONS ON SIGNAL PROCESSING* (2002–2006). He is currently an Associate Editor for the *IEEE TRANSACTIONS ON COMMUNICATIONS* and the *IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS, PART B*. He was the publications chair of ICASSP'07 and the technical program committee co-chair of ITW'07.