Scalable Data Multicast Using Expanding Window Fountain Codes

Dejan Vukobratović, Vladimir Stanković, Dino Sejdinović, Lina Fagoonee, and Zixiang Xiong

***Abstract*— Digital Fountain (DF) codes were introduced as an efficient and universal Forward Error Correction (FEC) solution for data multicast over lossy packet networks. However, in real-time applications, the DF encoder cannot make use of the "rateless" property as it was proposed in the DF framework, due to its delay constraints. In this scenario, many receivers might not be able to collect enough encoded symbols (packets) to perform succesful decoding of the source data block (e.g., they are connected as a low bit-rate receivers to a high bit-rate source stream, or they are affected by severe channel conditions). This paper proposes an application of recently introduced Expanding Window Fountain (EWF) codes as a scalable and efficient solution for real-time multicast data transmission. We show that, by carefully optimizing EWF code design parameters, it is possible to design a flexible DF solution that is capable of satisfying multicast data receivers over a wide range of data rates and/or erasure channel conditions.**

I. Introduction

After sparse graph codes and iterative decoding algorithms revolutionized the channel coding field, another idea based on these concepts made a breakthrough in communications theory: Digital Fountain (DF) codes. Conceptually introduced in [1], the DF framework is a universal capacity-approaching Forward Error Correction (FEC) solution for multicasting data, over lossy packet networks without feedback, to a set of receivers with different capabilities and erasure channel conditions. DF codes possess the "rateless" property, that is, they can generate potentially infinite amounts of encoded symbols given the input symbols of the source data block. The DF framework became a practical solution upon introduction of the first class of DF-compliant sparse graph codes: LT codes [2]. LT codes are able to provide a complete recovery of the transmitted data block, with high probability, for each receiver collecting any set of encoded symbols of total number slightly larger than the number of input symbols of the source data block. A degree distribution can be chosen such that the encoding/decoding complexity of LT codes is of the order $O(k \log k)$ for a source data block of length k input symbols. Raptor codes [3] are an improvement over LT codes, obtained by precoding an LT code defined by degree distribution of constant average value (instead of logarithmic in k) with a high-rate Low-Density Parity-Check (LDPC) code. Raptor codes represent a state-of-the-art DF solution with excellent performance and an encoding/decoding complexity of the order $O(k)$.

For delay-constrained applications, such as real-time multimedia streaming, the DF encoder cannot make use of the "rateless" property (as proposed in the original DF framework). Indeed, the DF encoder can produce "only" a finite amount of encoded symbols per information data block before moving to the next source block. In this scenario, many receivers might not be able to collect enough encoded symbols (per source block) to perform successful decoding of the whole source block (e.g., low bit-rate receivers connected to a high bit-rate source stream, or receivers affected by severe channel conditions). However, many delay-constrained applications do not require that each receiver recovers the entire source data block, but as much of the block as possible, because each decoded input symbol progressively increases the source data reconstruction quality. In addition, the importance of the input symbols of the source block is typically highest at the start of the block and decreases towards the end. These applications call for Unequal Error Protection (UEP) codes that associate different protection levels to the subsets of input symbols of different importance.

In this paper, we consider a scenario where a delay-constrained information source is multicasting data to a set of receivers divided into different receiver classes (based on available bandwidth and/or channel quality). We assume that the set of source input symbols is split into a number of importance classes, where the most important class defines the basic Quality of Service (QoS) guarantees, and subsequent importance classes define further QoS improvements. As an example of such a source model, we will be particularly interested in a scalable image or video coder. In this case, the source bitstream is divided into the base layer that represents the most important class, and a number of additional, so called enhancement layers, that, if received, progressively improve the received image/video quality. For a scalable source, and a number of different receiver classes, we propose a scalable DF multicast system that is able to accomodate a receiver class with the worst conditions with at least basic QoS guarantees, as well as allocating progressively increasing QoS guarantees for the receiver classes with succesively better conditions. Our solution is based on the UEP DF codes named Expanding Window Fountain (EWF) codes [4], which are appropriate for this scenario due to their

D. Vukobratović is with the Department of Communications and Signal Processing, University of Novi Sad, Novi Sad, Serbia. dejanv@uns.ns.ac.yu

V. Stanković is with the Department of Communication Systems, InfoLab 21, Lancaster University, Lancaster, UK. v.stankovic@lancaster.ac.uk

D. Sejdinović is with the Centre for Communications Research, Department of Electrical & Electronic Engineering, University of Bristol, Bristol, UK. d.sejdinovic@bristol.ac.uk

L. Fagoonee is with the Department of Communication Systems, InfoLab 21, Lancaster University, Lancaster, UK. l.fagoonee@lancaster.ac.uk

Z. Xiong is with the Department of Electrical and Computer Engineering, Texas A&M University, College Station, TX, USA. zx@ece.tamu.usa

design flexibility and excellent performance. We show that, by carefully optimizing EWF code design parameters, it is possible to devise a flexible DF solution that is capable of satisfying multicast data receivers with a wide range of data rates and/or erasure channel conditions.

The paper is organized as follows. Section II reviews the design and the basic properties of EWF codes. In Section III, we propose a scalable DF multicast solution based on EWF codes and introduce various possible design scenarios for a scalable DF multicast system. This system is demonstrated by an example in Section IV, where a special attention is given to a system design for a scalable multimedia information source, and the parameters of an EWF code are further optimized to provide optimal end-to-end system performance. In Section V, we provide simulation results that confirm numerical solutions obtained in Section IV. Finally, the paper is concluded with details of our future work in Section VI.

II. EXPANDING WINDOW FOUNTAIN (EWF) CODES

Standard LT and Raptor codes are Equal Error Protection (EEP) codes, because they place equal protection on input symbols from the source data block. Recently, DF codes designs with the UEP property have emerged [4][5]. In this section, we describe EWF codes [4] as a the centerpiece of the proposed scalable DF multicast solution.

EWF codes are a novel class of DF codes with an UEP property, based on the idea of "windowing" the data set. We assume that the data block to be transmitted consists of k input symbols (or information packets). Let the sequence of expanding windows (input symbol subsets), where each window is contained in the next one in the sequence, be defined over the information data block (see Figure 1). Using the defined set of expanding windows, EWF encoding proceeds in a slightly different fashion than the usual LT encoding, i.e., to create a new encoded symbol, one of the windows from the set is randomly selected with respect to a probability distribution defined over the set of expanding windows. Upon window selection, a new encoded symbol is determined with an LT code of suitably chosen degree distribution as if encoding were performed only on the input symbols from the selected window. This procedure is repeated at the DF encoder for each encoded symbol.

More formally, the EWF code $\mathscr{F}_{EW}(\Theta,\Gamma,\Omega^{(1)},\dots,\Omega^{(r)})$ can be defined using the set of polynomials $\Theta(x),\Gamma(x),\Omega^{(1)}(x),\dots,\Omega^{(r)}(x)$. Polynomial $\Theta(x) = \sum_{i=1}^{r}\Theta_i x^i$, where $\Theta_i = \frac{k_i}{k}$, describes the division of the information data block into the set of r expanding windows[1] of size $k_i, 1 \leq i \leq r$, where $k_i < k_j$ if $i < j$, and $k_r = k$. The probability distribution associated with the set of expanding windows is described using polynomial $\Gamma(x) = \sum_{i=1}^{r}\Gamma_i x^i$, where Γ_i is the probability of selecting the i-th window.

[1]For the rest of this paper, we will assume without loss of generality that one end of each expanding window is fixed at the beginning of the data block, i.e., the importance of data decreases from the begining towards the end of the data block.

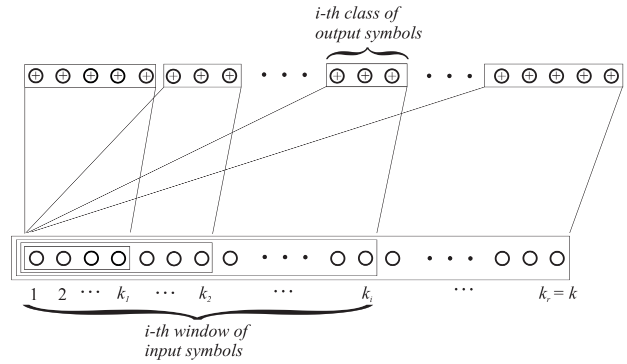


Fig. 1. Expanding Window Fountain (EWF) Codes.

The degree distribution $\Omega^{(j)}(x) = \sum_{i=1}^{k_j}\Omega_i^{(j)}x^i$ describes the LT encoding performed on the j-th window. To summarize, EWF code $\mathscr{F}_{EW}(\Theta,\Gamma,\Omega^{(1)},\dots,\Omega^{(r)})$ assigns each encoded symbol to the j-th window of size $k_j = \Theta_j \cdot k$ with probability Γ_j and encodes the data from the selected window using the LT code with the degree distribution $\Omega^{(j)}(x) = \sum_{i=1}^{k_j}\Omega_i^{(j)}x^i$.

By dividing the information data block using the set of expanding windows, we classify the set of input symbols into r disjoint importance classes. Input symbols belonging to the first (innermost) window are considered to be the most important, and they participate in forming each of the encoded symbols. The i-th importance class, $2 \leq i \leq r$, contains $s_i = k_i - k_{i-1} = (\Theta_i - \Theta_{i-1})k$ input symbols, i.e., the input symbols from the i-th window that do not belong to $(i-1)$-th window. Before presenting the main results of this paper, we define asymptotic erasure probability (i.e., the probability that an input symbol of the infinitely long EWF code is unknown after infinitely many iterations of iterative decoding) for input symbols of each importance class of EWF codes decoded using the iterative message-passing Belief-Propagation (BP) decoder. Evolution of the erasure probability for input symbols of each class with the iterations of the decoding algorithm is derived using the generalized and-or-tree analysis in [4]. We summarize the result in the following lemma (c.f. [4]):

Lemma 2.1: For EWF code $\mathscr{F}_{EW}(\Theta,\Gamma,\Omega^{(1)},\dots,\Omega^{(r)})$, the probability $y_{l,j}$ that the input node of the j-th importance class is not recovered at the receiver upon collecting $(1+\varepsilon)k$ encoded symbols, where ε is the reception overhead, after l iterations of the iterative decoder is:

$$y_{0,j} = 1,$$

$$y_{l,j} = \delta_j\left(1 - \sum_{i=j}^{r}\frac{\frac{\mu_i\Gamma_i}{\Theta_i}}{\sum_{m=j}^{r}\frac{\mu_m\Gamma_m}{\Theta_m}}\beta_i\left(1 - \sum_{m=1}^{i}\frac{\Theta_m - \Theta_{m-1}}{\Theta_i}y_{l-1,m}\right)\right), \tag{1}$$

where $\delta_j(x)$ and $\beta_j(x)$ are given by the expressions:

$$\delta_j(x) = e^{(1+\varepsilon)\sum_{l=j}^{r}\frac{\Gamma_l\mu_l}{\Theta_l}(x-1)}, \tag{2}$$

$$\beta_j(x) = \frac{\Omega'^{(j)}(x)}{\Omega'^{(j)}(1)}, \tag{3}$$

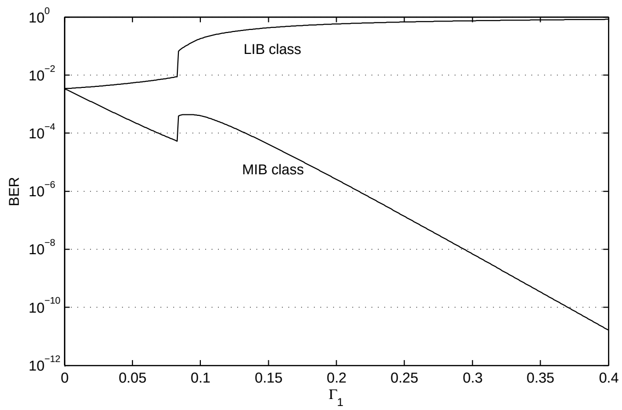


Fig. 2. Asymptotic analysis of Bit Error Rate (BER) versus Γ_1 for EWF codes (reception overhead is $\varepsilon = 0.05$).

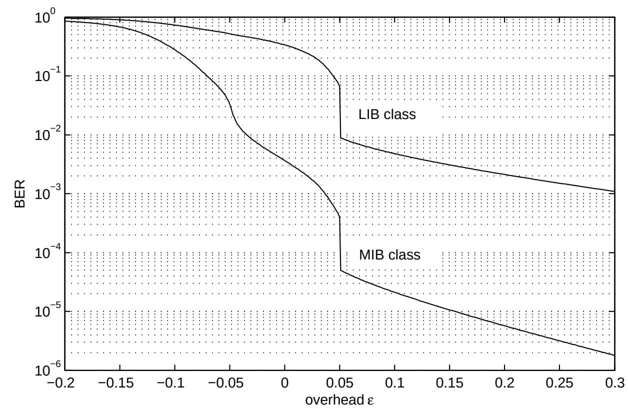


Fig. 3. Asymptotic analysis of BER versus overhead ε ($\Gamma_1 = 0.084$).

and $\mu_j = \sum_{i=1}^{k_j} i\Omega_i^{(j)}$ is the average degree of the encoded symbol generated from the input data of the j-th window.

It is useful to demonstrate possible applications of the previous lemma by a numerical example. Let the code under consideration be the EWF code $\mathscr{F}_{EW}(\Theta(x) = 0.1x + x^2, \Gamma(x) = \Gamma_1 x + (1-\Gamma_1)x^2, \Omega^{(1)}, \Omega^{(2)})$. Note that $r = 2$ and the degree distributions applied on both windows is the same and equal to degree distribution given in equation (5) in Section IV. Asymptotic erasure probabilities for Most Important Bit (MIB) class $y_{\infty,1}$ and Least Important Bit (LIB) class $y_{\infty,2}$, as a function of the first window selection probability Γ_1, are presented in Figure 2. Additionally, for $\Gamma_1 = 0.084$ (the position of a local minimum of $y_{\infty,1}$ in Figure 2), we can track asymptotic erasure probabilities of the MIB and LIB classes of source symbols as a function of overhead ε of encoded data collected at the receiver (Figure 3).

III. SCALABLE DF MULTICAST DATA TRANSMISSION

DF data 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 DF encoder and sent to the receivers. In practice, DF multicast is usually concerned with two problems. First, the amount of the encoded symbols sent is finite, moreover, for many delay-constrained applications including real-time multimedia streaming, the amount of encoded data that can be generated per source data block is severely limited. The second problem is the typical "avalanche decoding" behavior of the DF decoder (or the iterative BP decoder) at the receiver end; that is, with slightly less amount of received encoded symbols than needed for succesful decoding, the DF decoder is typically able to reconstruct only a negligible part of the data block transmitted.

In current DF multicast transmission systems [6], data multicast transmission typically proceeds in two phases. In the first phase, enough encoded packets are sent to facilitate successful decoding for most of the receivers. If some of the receivers cannot collect enough encoded data to finish decoding, by appropriate feedback signalling they indicate their participation in the second, so called 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 DF code is used and all the source data are given the same priority, regardless of the source characteristics, receiver bandwidths and channel conditions.

In this work, we develop a scalable DF multicast scheme that takes into account both, the limited amount of encoded data that can be produced per transmitted data block, and the variability in the (erasure) channel parameters and conditions for different receivers. In other words, we adapt encoded symbols at the output of the DF encoder to importance of the data contained in the information block at the input. This adaptation is performed in such a manner that ensures performance guarantees, in terms of the amount and importance of received data, for each receiver (or class of receivers) in the DF system. In the following, we describe our DF multicast system solution in detail.

A. Scalable DF Multicast: System Setting

We assume that a DF encoder generates and sends data to sets of receivers of a multicast session over a lossy packet network. The DF encoder transforms the delay-constrained information data stream, segmented into data blocks of length k input symbols, into an encoded data stream of length $(1+\varepsilon_S)k$ encoded symbols, where ε_S is the source overhead that saturates the available data rate of the multicast connection. The encoded data stream is sent to sets of multicast receivers with different capabilities over a lossy packet network. We quantify each receiver's capability by its reception overhead ε_R, taking into account both the data rate of the receiver multicast connection and the (packet) erasure channel probability between the source and the receiver. In other words, the receiver with reception overhead $\varepsilon_R \leq \varepsilon_S$ will be able to receive $(1+\varepsilon_R)k$ encoded symbols out of the total $(1+\varepsilon_S)k$ sent symbols, per transmitted data block. In the following, we assume without loss of generality that the DF multicast system serves r different receiver classes ordered by their reception overheads: $\varepsilon_{R,1} < \varepsilon_{R,2} < \ldots < \varepsilon_{R,r} \leq \varepsilon_S$. It is worth noting that, although our system is set up in terms of overheads, it is straightforward to translate it to the data rate system representation.

The task of the DF encoder is to supply receivers with information data blocks through DF encoded data streams. We assume that the input symbols of information data blocks are such that their importance decreases from the beginning of the block towards its end (e.g., the information stream is the output of a scalable image or video coder). Due to a different reception capabilities of different receivers, we want to match the DF encoded data stream to the appropriate receiver classes; that is, the first receiver class (with the worst reception conditions) would recover the first part (the most important part) of the source data with high probability, the second receiver class would be able to recover the first two parts of the source data with high probability, etc. This calls for an EWF code with expanding windows defined to match the importance classes of the source data. In order to match the code to each of the r receiver classes, we select EWF code $\mathscr{F}_{EW}(\Theta,\Gamma,\Omega^{(1)},\dots,\Omega^{(r)})$ with r expanding windows defined across each source block to be transmitted. EWF encoding with finite source overhead ε_S is then applied at the source, as described in Section II.

For a given reception overhead of a receiver of the i-th receiver class, $\varepsilon_{R,i}$, and the parameters of selected EWF code $\mathscr{F}_{EW}(\Theta,\Gamma,\Omega^{(1)},\dots,\Omega^{(r)})$, using *Lemma* 2.1 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. From *Lemma* 2.1, it follows that $p_i^{(j)} = f(i,\varepsilon_{R,j},\mathscr{F}_{EW}(\Theta,\Gamma,\Omega^{(1)},\dots,\Omega^{(r)}))$. Using $p_i^{(j)}$, and under the asymptotic assumption of independence of probabilities $p_i^{(j)}$ for different input symbols, we can calculate the probability $P_i^{(j)}$ that the i-th importance class of the input symbols is completely reconstructed by the j-th receiver class:

$$P_i^{(j)} = (1-p_i^{(j)})^{s_i}, \tag{4}$$

where s_i is the number of input symbols in the i-th importance class.

We will use the set of probabilities $P_i^{(j)}$ to define QoS guarantees[2] for each receiver class of the proposed scalable DF multicast system. It is worth noting that $P_i^{(j)} < P_i^{(k)}$ for $j<k$ due to *Lemma* 2.1 and $\varepsilon_{R,j} < \varepsilon_{R,k}$; that is, a receiver in the k-th class will be able to satisfy all the QoS guarantees imposed on the j-th receiver class. Therefore, it is convenient to define QoS guarantees for the scalable DF 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 of less importance, the i-th receiver class is not provided with any QoS guarantees.

[2]Note that the value $P_i^{(j)}$ defines the percentage of the receivers in the j-th receiver class that will be able to reconstruct the set of all input symbols from the i-th importance class.

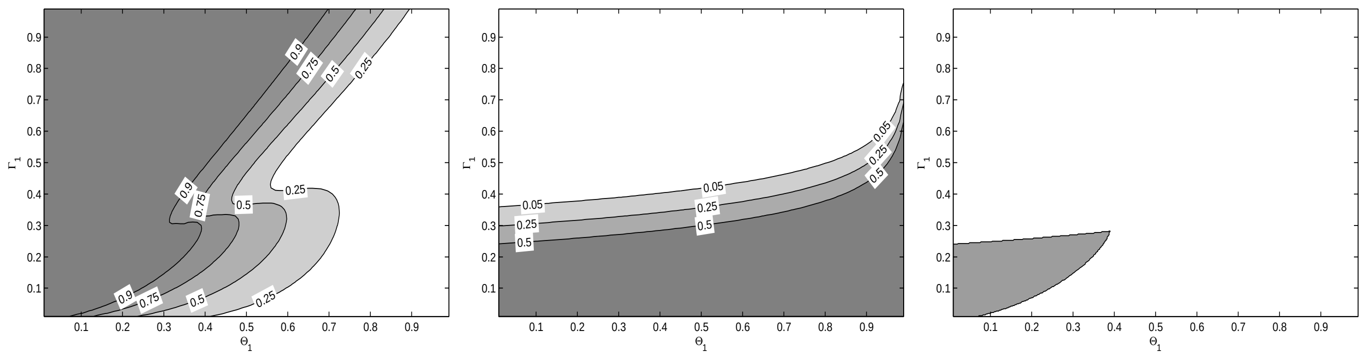
B. Scalable DF Multicast: Design Scenarios

Using the previously described system setting, the scalable DF multicast system design reduces to the design of the EWF code such that given QoS guarantees for different receiver classes are satisfied. However, due to a large number of parameters included in this design process, it is possible to define many design scenarios of practical interest. For the design of the scalable DF multicast system, we identify three sets of "system parameters" of interest: the set of QoS guarantees $\{P_1^{(1)},P_2^{(2)},\dots,P_r^{(r)}\}$, shortly denoted by $P(x)=\sum_{i=1}^r P_i^{(i)}x^i$, the set of reception overheads $\{\varepsilon_{R,1},\varepsilon_{R,2},\dots,\varepsilon_{R,r}\}$ of different receiver classes, denoted by $\varepsilon(x)=\sum_{i=1}^r \varepsilon_{R,i}x^i$, and the EWF code division of the transmitted data block into importance classes, described by $\Theta(x)=\sum_{i=1}^r \Theta_i x^i$. By fixing two out of three of these system polynomials, we can optimize the third system polynomial by appropriately using remaining EWF code design parameters in the optimization procedure[3]. In the following, we discuss some design options.

- *Scenario 1* - $\Theta(x)=f(\varepsilon(x),P(x))$: In this scenario, the inputs of the design process are system polynomials $\varepsilon(x)$ and $P(x)$, and we are interested in optimizing $\Theta(x)$. In other words, for a given reception performance and requested QoS guarantees, how do we optimally partition the source data into the importance classes?
- *Scenario 2* - $P(x)=f(\varepsilon(x),\Theta(x))$: This scenario optimizes $P(x)$ given system polynomials $\varepsilon(x)$ and $\Theta(x)$. Therefore, for a given reception performance and division into importance classes, we are interested in the maximum QoS guarantees the DF system can offer?
- *Scenario 3* - $\varepsilon(x)=f(P(x),\Theta(x))$: In the last scenario, we fix $\Theta(x)$ and $P(x)$, and we optimize $\varepsilon(x)$. This means that, given the division into importance classes and QoS guarantees requested for each importance class, we are interested in reception performance needed for each of the receiver classes. In case when the reception performance is dominated by data rate of the receivers (i.e., a packet network with negligible packet losses), this scenario determines the minimum data rates needed by each receiver class in the system to meet QoS guarantees $P(x)$ for different importance classes $\Theta(x)$.

The previous design scenarios assume that the EWF code parameters except $\Theta(x)$, i.e., $\Gamma(x)$ and the set of r degree distributions $\Omega^{(i)}(x)$, are not fixed in advance. We will use these parameters as additional "degrees of freedom" that are available for tuning the performances of the designed DF system. In the following section, we propose a design example for the first design scenario of the proposed scalable DF multicast system. Similar design examples for other scenarios can be easily extended from this simple example.

[3]Note that the system polynomials are dependent, e.g., $P(x) = f(\Theta(x),\varepsilon(x))$.



(a) $P_1^{(1)} = f(\Gamma_1, \Theta_1)$ (b) $P_2^{(2)} = f(\Gamma_1, \Theta_1)$ (c) Admissible (Γ_1, Θ_1) region

Fig. 4. The region of (Γ_1, Θ_1) that satisfies given constraints $\varepsilon(x)$ and $P(x)$.

IV. SCALABLE DF MULTICAST: NUMERICAL SOLUTION

In this section, we are interested in the first design scenario: determining the division $\Theta(x)$ of source data into importance classes that meets the requested QoS guarantees $P(x)$, for a given reception performance $\varepsilon(x)$. Due to the fact that equation (1) is a non-linear recursion formula and that (asymptotic) erasure probabilites $p_i^{(i)}$ are output of this formula after infinitely many iterations, we provide only a numerical solution for this design problem.

A. Scalable DF Multicast with Two Receiver Classes

For simplicity, we assume a setting with $r = 2$ receiver classes (thereby, there are two windows of the EWF code). We additionally simplify the EWF code design by assuming that on both windows we apply the same degree distribution (constant average degree distribution proposed in [3], and used in the UEP DF code design [4][5]):

$$\begin{aligned}\Omega^{(1)}(x) &= \Omega^{(2)}(x) = 0.007969x + 0.493570x^2 + \\ &+ 0.166220x^3 + 0.072646x^4 + 0.082558x^5 + \\ &+ 0.056058x^8 + 0.037229x^9 + 0.055590x^{19} + \\ &+ 0.025023x^{65} + 0.003135x^{66} \qquad (5)\end{aligned}$$

Using these simplifications, the design of the EWF code $\mathscr{F}_{EW}(\Theta_1 x + x^2, \Gamma_1 x + (1-\Gamma_1)x^2, \Omega^{(1)}, \Omega^{(2)})$ is determined by two independent variables: Γ_1 and Θ_1 (the probability of selection of the first window and the fraction of the data contained in it). The design problem can be stated as follows: for a given reception overheads $(\varepsilon_{R,1}, \varepsilon_{R,2})$ and requested QoS guarantees $(P_1^{(1)}, P_2^{(2)})$, find the set of all values of Θ_1 (with their corresponding values of Γ_1). In general, as a result of this design process, we obtain a set (region) of possible (Θ_1, Γ_1) pairs that satisfies given conditions. Note that, depending on the values of $\varepsilon(x)$ and $P(x)$, this set can be empty, providing no solution for the requested scenario.

As an example, we select the following constraints: $\varepsilon(x) = (0.25, 1)$ and $P(x) = (0.9, 0.5)$ and the data block length of $k = 4250$ symbols (packets). We adopt, without loss of generality, that the symbol size is 8 bytes[4]. We have two classes of receivers: the first, worse class, characterized by the 25% reception overhead, and the second, better class, with 100% overhead[5]. The QoS guarantees require that a receiver in the worse class has a probability of reconstruction of more important data block of at least 90% (i.e., at least 90% of receivers in the worse class will reconstruct more important data block entirely), while a receiver in the better class should, in addition, be able to reconstruct the less important data block with probability of at least 50%. The reconstruction probabilities of the more important block for the worse class of receivers, $P_1^{(1)}$, and the less important block for the better class of receivers, $P_2^{(2)}$, are given as functions of two variables (Θ_1, Γ_1) in Figures 4(a) and 4(b), respectively. The darkest shaded region on Figures 4(a) and 4(b) consists of (Θ_1, Γ_1) pairs that satisfy given QoS constraints, $P_1^{(1)} > 0.9$ and $P_2^{(2)} > 0.5$, respectively. Additionally, on both figures, one can track changes of probabilities $P_1^{(1)}$ and $P_2^{(2)}$ presented by lighter shaded gray regions. The intersection of the darkest shaded regions, presented in Figure 4(c), is the region of (Θ_1, Γ_1) pairs that satisfy given constraints $\varepsilon(x)$ and $P(x)$.

Figure 4(c) is a solution of a given design scenario. Since its output is a set of (Θ_1, Γ_1) pairs, our next task is to select operational pair (Θ_1, Γ_1) out of the given region using a suitable criterion. One way to proceed would be to select a solution that maximizes Θ_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 $(\Theta_1, \Gamma_1) = (0.39, 0.2825)$ that treats 39% of the transmitted data block as the more important data. However, other optimality considerations of points in the (Θ_1, Γ_1) region are possible, particularly in the case when the information source is a scalable image or video coder. We provide some examples in the following subsection.

[4] Data block length and symbol size are selected to fit exactly the size of the JPEG2000 512×512 Lena image of the highest compression rate used.

[5] For example, this corresponds to a DF multicast setting where the source stream of 1Mbit/s bit rate is transmitted to two different receiver classes, with 1.25Mbit/s and 2Mbit/s of available bandwidth respectively, assuming that the DF encoded multicast data rate is at least 2Mbit/s, and that the packet network is lossless.

B. Optimal (Θ_1, Γ_1) Pair Selection

The design example from the previous subsection can be applied to any kind of information source. As a result, the set of (Θ_1, Γ_1) pairs is obtained that provide certain QoS performance for each class of receivers, in terms of the reconstruction probabilities of different importance classes. For a specific case of a scalable image/video information source, different points from the (Θ_1, Γ_1) region will have different performance in terms of the quality of the reconstructed image/video data. In the following, we will focus on a scalable image/video source and discuss the possibilities of further optimization of EWF codes in this scenario.

Latest solutions for multimedia distribution often rely on scalable image coders, such as SPIHT [7] or JPEG2000 [8], or scalable video coders, such as three-dimensional (3D) scalable wavelet-based video coders [9][10]. Scalable image/video coders are particularly useful in multicast scenarios, due to the fact that they efficiently accomodate receivers with different data rates and/or channel conditions. The output bitstream of a scalable image/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 image/video quality. This makes a scalable image/video coder together with a scalable DF multicast system based on EWF codes a promising combination for multicast multimedia distribution services. Since EWF codes are a UEP DF 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 such that we improve the received image/video quality at the receiving end provides a powerful DF Joint Source-Channel Coding (DF-JSCC) multicast solution.

To select "optimal" (Θ_1, Γ_1) point from the region obtained as an output of the DF multicast system design, we apply two different approaches: the optimization with respect to the expected number of correctly received information symbols and the optimization with respect to the expected (image/video) distortion at the receiving end. These performance criteria are refered to as rate-based optimization and distortion-based optimization, respectively [11]. Both of these performance criteria take into account the fact that the reconstruction process at a scalable image/video decoder deteriorates significantly after the first transmission error is encountered, due to an error propagation effect. Therefore, for most scalable image/video decoders, the process of source data reconstruction terminates upon detecting the first undecoded symbol.

For the definition of the rate-based and distortion-based optimization criteria, we will assume that the transmitted data block is divided into N segments (layers) of lengths $L_1, L_2, \ldots, L_N$ symbols. The importance of data contained in segments decreases from the first towards the last segment in the block. Reconstruction of the transmitted data block at the receiver is based on correctly received consecutive segments until the first segment for which a channel transmission error is detected. We denote a probability of correct reconstruction of each of N data segments as $P_1, P_2, \ldots, P_N$, respectively.

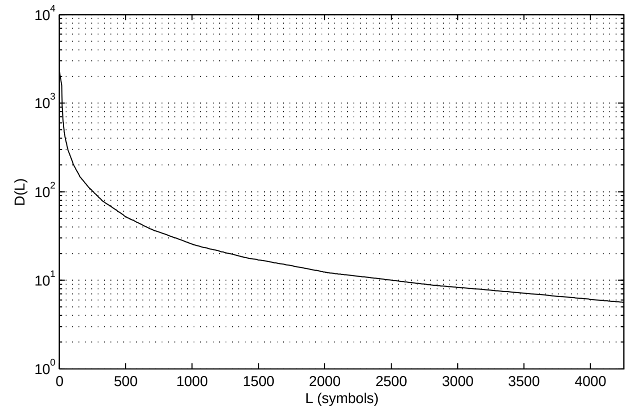


Fig. 5. Rate-distortion function of the JPEG2000 coded Lena image.

The transmission scheme that maximizes the average number of correctly received segments (i.e., the average length of correctly received consecutive information symbols in the data block) at the receiver is considered to be rate-based optimal. This refers to a scheme that maximizes the following expression:

$$L_{avg} = \sum_{i=0}^{N} P(i) \cdot L(i), \tag{6}$$

where $L(i)$ is the total length (in information symbols) of the first i data segments:

$$L(i) = \begin{cases} 0 & \text{for } i = 0 \\ \sum_{j=1}^{i} L_j & \text{for } 0 < i \leq N, \end{cases} \tag{7}$$

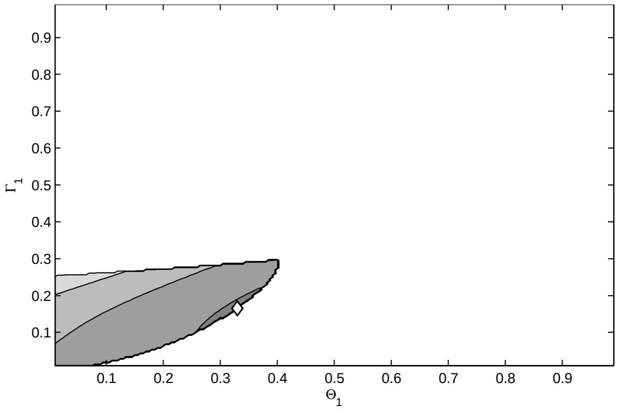
and $P(i)$ is the probability that the first i consecutive segments are correctly received:

$$P(i) = \begin{cases} 1 - P_1 & \text{for } i = 0 \\ \prod_{j=1}^{i} P_j \cdot (1 - P_{i+1}) & \text{for } i = 1, 2, \ldots, N-1 \\ \prod_{j=1}^{N} P_j & \text{for } i = N. \end{cases} \tag{8}$$

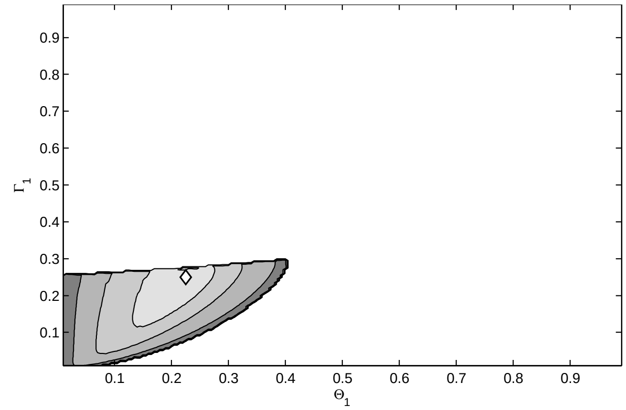
The motivation for using rate-based optimization lies in the fact that if a longer bitstream is used for reconstruction, a better reconstruction quality will be obtained. Hence, it is desirable to delay the first uncorrected error as much as possible. Note that the rate-based optimization is independent of the information source and the data transmitted.

In order to define the distortion-based optimization criterion, we need a rate-distortion function $D(L)$, which is a measure of image/video distortion after reconstruction based on the first L information symbols. In this example we use operational rate-distortion function $D(L)$ of JPEG2000 coder [8] derived from the 512×512 Lena image (Figure 5). It is worth noting that the operational $D(L)$ function is a function of both, the type of a scalable coder used and the content of the data transmitted. Therefore, the same holds for the distortion-based criterion.

The transmission scheme that minimizes the expected distortion of the image reconstructed at the receiver is



(a) Rate-based optimal point $(\Theta_1^{(r.o)}, \Gamma_1^{(r.o)})$.



(b) Distortion-based optimal point $(\Theta_1^{(d.o)}, \Gamma_1^{(d.o)})$.

Fig. 6. The optimal (Θ_1, Γ_1) points for different optimization criteria.

considered to be distortion-based optimal. It is obtained by minimizing the following expression:

$$D_{avg} = \sum_{i=0}^{N} P(i) \cdot D(L(i)), \tag{9}$$

where $L(i)$ and $P(i)$ are the same length and probability parameters as in the rate-based optimization given by equations (7) and (8), respectively, D(0) is the source variance, and for $L > 0$, $D(L)$ is the operational rate-distortion function of the image/video coder.

We apply the rate-based and distortion-based optimization on the setting from the previous subsection in order to find the optimal (Θ_1, Γ_1) point from the region given in Figure 4(c). Again, we assume that the transmitted data block is divided into $N = r = 2$ segments of lengths $L_1 = s_1 = \Theta_1 k$ and $L_2 = s_2 = (\Theta_2 - \Theta_1)k$. The probabilities of the correct reconstruction of each data segment for a receiver in the i-th receiver class, $i = \{1, 2\}$, are given as $P_1 = P_1^{(i)}$ and $P_2 = P_2^{(i)}$.

In the multicast scenario, we are dealing with a number of receiver classes, hence the expressions for L_{avg} and D_{avg} have to be averaged over all the classes. We denote by $L_{avg}^{(i)}$ the average length of correctly received data at the receiver of the i-th receiver class. The rate-based optimal solution is the point $(\Theta_1^{(r.o)}, \Gamma_1^{(r.o)})$ that maximizes average number of received segments across both receiver classes:

$$L_{avg} = \frac{1}{N} \sum_{i=1}^{N} L_{avg}^{(i)} = \frac{L_{avg}^{(1)} + L_{avg}^{(2)}}{2}. \tag{10}$$

Similarly, the distortion-based optimal solution in the multicast scenario is the point $(\Theta_1^{(d.o)}, \Gamma_1^{(d.o)})$ that minimizes the average image distortion across receiver classes:

$$D_{avg} = \frac{1}{N} \sum_{i=1}^{N} D_{avg}^{(i)} = \frac{D_{avg}^{(1)} + D_{avg}^{(2)}}{2}, \tag{11}$$

where $D_{avg}^{(i)}$ is the expected distortion of reconstructed data at the receiver of the i-th receiver class.

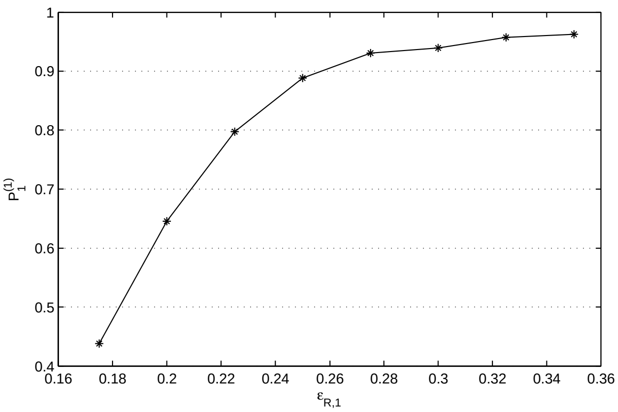
For all (Θ_1, Γ_1) pairs in the admissible region of Figure 4(c), we calculated L_{avg} and D_{avg} values using equations (10) and (11), respectively. The results are presented in Figures 6(a) and 6(b), respectively, where the values of L_{avg} and D_{avg} increase towards the darkest shaded regions. The rate-based optimal coordinates, attained for the point $(\Theta_1^{(r.o)}, \Gamma_1^{(r.o)}) = (0.33, 0.165)$, maximize L_{avg} value equal to $L_{avg}(0.33, 0.165) = 2547$ information symbols. Therefore, the number of symbols that is expected to be correctly received on average by both receiver classes reaches a maximum value of 2547 symbols (163000 bits), out of a total of 4250 symbols (272000 bits).

In order to compare optimization results, we applied the distortion-based optimization on the information block containing 512×512 JPEG2000 coded Lena image. For the distortion-based optimization, we obtained the optimal minimum D_{avg} point $(\Theta_1^{(d.o)}, \Gamma_1^{(d.o)}) = (0.225, 0.25)$ that has an average distortion value of $D_{avg}(0.33, 0.22) = 23.15$. For the rate-based optimal solution $(\Theta_1^{(r.o)}, \Gamma_1^{(r.o)}) = (0.33, 0.165)$, an average distortion is equal $D_{avg}(0.33, 0.165) = 127$.

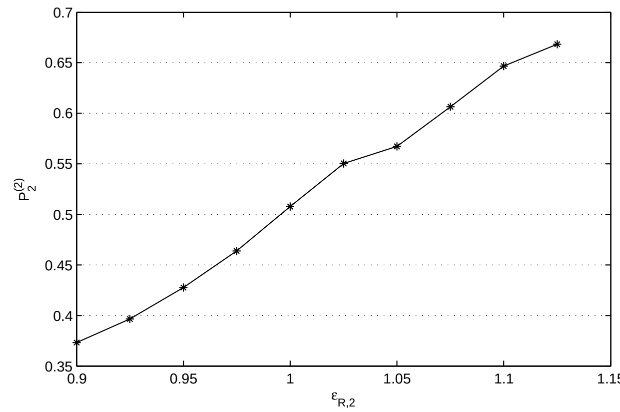
We observed a considerable variablity of the average distortion inside the region. Indeed, even the rate-based optimized solution, which approximates the rate-distortion function with a linear function, is away from the global minimum found by the distortion-based optimization. Hence, the distortion-based optimization is needed to ensure a proper (Θ_1, Γ_1) pair selection. We note that the result of the distortion-based optimization is shorter and better protected most important class, which is a typical behavior for scalable sources [11].

V. SCALABLE DF MULTICAST: SIMULATION RESULTS

The numerical results presented in the previous section are derived using asymptotic erasure probabilities of input symbols, but applied on the finite-length ($k = 4250$) DF system scenario. To verify that these solutions are good approximation of the "real-world" results, we perform simulation experiments. We select a simulation setting that corresponds to the numerical solution presented in the Section IV A, where the optimal point $(\Theta_1, \Gamma_1) = (0.39, 0.2825)$ is



(a) $P_1^{(1)} = f(\varepsilon_{R,1})$.



(b) $P_2^{(2)} = f(\varepsilon_{R,2})$.

Fig. 7. Simulation results for reconstruction probabilities versus reception overheads: $P_1^{(1)} = f(\varepsilon_{R,1})$ and $P_2^{(2)} = f(\varepsilon_{R,2})$.

selected from the region in Figure 4(c) using the criterion of maximizing the amount of data placed in the most important window (i.e., maximizing Θ_1 value over the region). Therefore, we simulated the performance of the EWF code with parameters $\mathscr{F}_{EW}(0.39x+x^2, 0.2825x+0.7175x^2, \Omega^{(1)}, \Omega^{(2)})$, the information block length of $k = 4250$ symbols, and the degree distributions in polynomial form as (5).

For the first receiver class, we simulate the average probability of correct reconstruction of the most important class of input symbols, $P_1^{(1)}$, for different reception overheads in the interval around the value $\varepsilon_{R,1} = 0.25$. For the second receiver class, similar simulations are performed for the probability $P_2^{(2)}$ in the interval around the reception overhead $\varepsilon_{R,2} = 1$. The results are presented in Figure 7(a) and 7(b), respectively.

We note a very good matching of the experimental results presented in Figure 7 with the results obtained in Section IV, for the modest information block length of $k = 4250$ bits (symbols). Although applied on the finite-length realistic scenario, the simulation results confirm our analysis based on the asymptotic probability expressions and the numerical solutions obtained in the previous two sections.

VI. CONCLUSIONS

A novel, scalable DF multicast system, based on EWF codes is proposed as an effective solution for real-time data delivery to various classes of receivers with different reception conditions. In this scenario, classical EEP DF codes would perform poorly due to a potentially large number of receivers that are not able to collect enough encoded symbols to perform successful decoding. A flexible EWF code design is shown to be a promising solution, where the EWF encoder adapts its encoded data stream to satisfy QoS guarantees offered to each receiver class. Although the proposed system is applicable to any information source, a scalable image/video coder is particularly suitable and effective in combination with an EWF encoder. We demonstrated that, in this case, further optimization is possible in order to adapt EWF codes to increase image/video reception quality.

In this paper, we presented basic ideas of applying EWF codes in a scalable DF multicast scenario. For our future work, we plan to investigate in more details the potential of the proposed scheme in the real-world scalable video distribution scenarios. Additionally, we would like to perform more detailed investigations on the benefits of precoding the EWF codes by high-rate outer LDPC codes (the Raptor code scenario) for the scalable DF multicast.

REFERENCES

[1] J. Byers, M. Luby, M. Mitzenmacher, and A. Rege, “A digital fountain approach to reliable distribution of bulk data,” *ACM SIGCOMM 98*, pp. 56–67, Vancouver, Canada, September 1998.

[2] M. Luby, “LT Codes,” *Proc. of the 43rd Annual IEEE Symp. Foundations of Computer Science (FOCS)*, pp. 271–282, Vancouver, Canada, November 2002.

[3] A. Shokrollahi, “Raptor Codes,” *IEEE Trans. on Information Theory*, vol. 52, No. 6, pp. 2551–2567, June 2006.

[4] D. Sejdinović, D. Vukobratović, A. Doufexi, V. Šenk, and R. Piechocki, “Expanding Window Fountain Codes for Unequal Error Protection,” *preprint, submitted for publication, 41st Annual Asilomar Conference on Signals, Systems and Computers 2007.*

[5] N. Rahnavard, B. N. Vellambi, and F. Fekri, “Rateless Codes with Unequal Error Protection Property” *IEEE Trans. on Information Theory*, vol. 53, No. 4, pp. 1521–1532, April 2007.

[6] T. Gasiba, T. Stockhammer, and W. Xu, “Reliable and Efficient Download Delivery with Raptor Codes,” *Proc. of the 4th International Symposium on Turbo Codes and Related Topics*, Munich, Germany, April 2006.

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

[8] D. Taubman and M. Marcellin, *JPEG2000: Image Compression Fundamentals, Standards and Practice*, Kluwer, 2001.

[9] J. Ohm, “Advances in scalable video coding,” *Proceedings of the IEEE,* vol. 93, pp. 42–56, January 2005.

[10] L. Luo, F. Wu, S. Li, Z. Xiong, and Z. Zhuang, “Advanced motion-threading techniques for 3-D wavelet video coding,” *Signal Processing: Image Communication*, vol. 19, pp. 601–616, August 2004.

[11] R. Hamzaoui, V. Stanković, and Z. Xiong, “Optimized Error Protection of Scalable Image Bit Streams,” *IEEE Signal Processing Magazine*, vol. 22, No. 6, pp. 91–107, November 2005.