

Integrating Non-Repetitive LT Encoders With Modified Distribution to Achieve Unequal Erasure Protection

Kuo-Kuang Yen, Yen-Chin Liao, Chih-Lung Chen, John K. Zao, *Senior Member, IEEE*, and Hsie-Chia Chang

Abstract—The performance of LT code is highly related to the code length. A decoder is more likely to deplete degree-1 encoding symbols and terminate during early stage when the code length is short. In this work, we modify the robust Soliton distribution (RSD) and increase the degree-1 proportion. More degree-1 encoding symbols can be generated to relieve early decoding termination. The proportion of low degrees, except for degree-1, is also reduced. Therefore, receivers collect less encoding symbols carrying redundant information. In addition, *Non-Repetitive (NR)* encoding scheme is proposed to avoid producing repeated degree-1 encoding symbols. To improve video transmission quality, previous studies redesign LT codes to provide Unequal Error Protection (UEP) for different Scalable Video Coding (SVC) layers. Unlike those studies to modify the code structure, we integrate multiple NR encoders to achieve UEP ability. Experimental results show that our UEP scheme outperforms previous studies in terms of the PSNR.

Index Terms—BP decoding, LT code, degree, non-repetitive encoding.

I. INTRODUCTION

APPLICATION layer forward error correction (ALFEC) has been introduced to improve multimedia communication quality in recent years [1]–[3]. ALFEC includes not only traditional block erasure codes, such as Reed-Solomon, Turbo and low-density parity check codes [4]–[6] but also fountain codes [7], [8]. Due to fixed code rates, the block erasure codes will waste bandwidth when channel condition is better than expected or fail to recover transmitted data when channel condition is underestimated. On the contrary, a fountain encoder potentially generates infinite encoding symbols. Original data is decodable once a receiver collect sufficient encoding symbols. Such rateless property allows fountain codes to approach various channel capacity simultaneously. Besides, low decoding

and encoding complexity also makes fountain codes suitable for ALFEC.

LT code is the first realization of the fountain codes [9]. Many multicast applications have employed LT codes for packet protection [10]–[12]. When generating an encoding symbol, an LT encoder first decides on the degree d of this symbol based on a degree distribution. The robust Soliton distribution (RSD) is widely used for the LT encoding. Then d out of all k input symbols are selected randomly and uniformly as neighbors of this encoding symbol. The bitwise exclusive-or result of these neighbors is transmitted as an encoding symbol. The number k is called *code length*. Each decoder employs the belief propagation (BP) algorithm to recover input symbols from $(1 + \epsilon)k$ received encoding symbols, where ϵ is the overhead. At each iteration, a BP decoder assigns the value of each degree-1 encoding symbol to its corresponding input symbol that is then decoded. Connections to the decoded input symbols are removed to produce the next batch of degree-1 encoding symbols. This process terminates if degree-1 encoding symbols are no longer available or input symbols are decoded completely. Raptor codes further improve the LT code performance and achieve linear encoding and decoding complexity [13]. Applications and standards have included the raptor codes, such as DVB-H standard for IPTV applications and Multimedia Broadcast Multicast Services (MBMS) of the Third Generation Partnership Project (3GPP). Combinations of scalable video coding (SVC) [10] and fountain codes with unequal error protection (UEP) property can also be found in [14], [15], which provide more flexibility to satisfy various applications.

The LT codes and raptor codes perform well when the code length k is on the order of 10^4 or longer [9], [13]. However, as k decreases, the average number of degree-1 encoding symbols becomes smaller during BP decoding process. A decoder will be more likely to exhaust degree-1 encoding symbols prematurely, leading to frequent decoding termination. Studies have focused on improving short-length LT codes. For example, sliding-window encoding was proposed in [16] to virtually extend the code length. This technique requires extra encoding symbol buffers and longer decoding time. Moreover, a new design of degree distribution proposed in [17] extends virtual code length as well. Unlike the previous studies, we enhance the performance of LT codes with short code length by reducing the probability of early decoding termination.

The first contribution of this paper is that we reduce symbol loss probability of LT codes. This is achieved by preventing BP decoding process from terminating at an early stage. The de-

Manuscript received August 07, 2012; revised December 25, 2012; accepted February 14, 2013. Date of publication June 19, 2013; date of current version November 13, 2013. This work was supported by the National Science Council of Taiwan, R.O.C., under grant NSC 101-2220-E-009-060 and NSC 101-2628-E-009-013-MY3. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Pal Halvorsen.

K.-K. Yen, Y.-C. Liao, C.-L. Chen, and H.-C. Chang are with the Department of Electronics Engineering and Institute of Electronics, National Chiao Tung University, Hsinchu, Taiwan (e-mail: dannies@oasis.ee.nctu.edu.tw; ycliaoee92g@gmail.com; lung@si2lab.org; hcchang@mail.nctu.edu.tw).

J. K. Zao is with the Department of Computer Science, National Chiao Tung University, Hsinchu, Taiwan (e-mail: jkzao@cs.nctu.edu.tw).

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.2013.2269898

TABLE I
NOTATIONS AND THEIR DEFINITIONS

k	: Number of input symbols.
ϵ	: Overhead.
ρ	: Number of decoded input symbols.
ρ/k	: Ratio of decoded input symbols.
$d(c_l)$	: Degree of encoding symbol c_l .
C_r	: A set of $(1 + \epsilon)k$ received encoding symbols.
$\Lambda(c_l)$	: A set of input symbols connected by encoding symbol c_l (neighbors of encoding symbol c_l).
$\Lambda(i_l)$	: A set of encoding symbols connecting to input symbol i_l (neighbors of input symbol i_l).
$R(\rho)$	: Average number of degree-1 encoding symbols during BP decoding process when ρ input symbols are decoded.
Ω^r	: Robust Soliton distribution (RSD).
$\tilde{\Omega}$	: Modified robust Soliton distribution for NR encoding scheme.
$\Gamma(\rho)$	: Occurrence probability of BP decoding termination when ρ input symbols are decoded.
$\Pi(\rho)$	: Cumulative probability of decoding termination. ($\Pi(\rho) = \sum_{q=0}^{\rho} \Gamma(q)$.)
S	: Successful decoding probability. (Probability of decoding all k input symbols from $(1 + \epsilon)k$ received encoding symbols, where $S = \Gamma(k) = 1 - \sum_{q=0}^{k-1} \Gamma(q) = 1 - \Pi(k-1)$.)
L	: Symbol loss probability. (Probability that an input symbol remains undecoded after BP decoding process, where $L = \sum_{\rho=0}^k \frac{k-\rho}{k} \Gamma(\rho)$.)
P_e	: Channel erasure probability.
$\hat{P}_e$	: Channel erasure probability of the worst case scenario with regard to channel conditions.
$\hat{Q}$	: Design parameter below which the probability of receiving repeated degree-1 encoding symbols must be guaranteed.
$\hat{\Omega}_1$	: Upper bound of the degree-1 proportion corresponding to $\hat{P}_e$ and $\hat{Q}$.
Δ	: Average number of symbol operations to generate an encoding symbol. (A symbol operation is either an exclusive-or between two symbols or a copy of one symbol to another.)

gree-1 proportion in RSD is increased. Receivers then collect more degree-1 encoding symbols to relieve decoding termination within the range $0 \leq \rho \leq k/2$. Besides, *Non-Repetitive* (NR) encoding scheme is introduced to avoid generating repeated degree-1 encoding symbols. The idea is to force the first k degree-1 encoding symbols to connect to different input symbols. Lastly, the degree distribution is further modified to reduce encoding symbols carrying redundant information.

The second contribution is that we integrate multiple NR encoders to provide UEP. Our scheme is compared with UEP schemes proposed in [14], [15], [17]. These schemes allow encoding symbols to connect to input symbols with different importance. On the contrary, each NR encoder in our UEP scheme generates encoding symbols from input symbols with the same importance. NR encoders can then be designed independently, which simplifies the design flow of our UEP scheme. In addition, the peak signal-to-noise ratio (PSNR) performance is evaluated between original and reconstructed videos of 18000 frames, or equivalently 10 minutes in length, instead of repeatedly transmitting the first group of pictures (GOP) as in [17].

The rest of this paper is organized as follows. We introduce LT code and its drawbacks with short code length in Section II. Section III illustrates the modification of RSD and NR encoding scheme. The proposed UEP scheme is described in Section IV. Then Section V compares our encoding scheme with the previous UEP schemes. This paper ends with a conclusion in Section VI.

II. BACKGROUND

A. LT Code

An LT code can be represented as $LT(k, \Omega)$, where k is the number of input symbols and Ω is a degree distribution, which specifies the proportion between different degrees as defined in the following.

Definition 1: A degree distribution is a probability set $\Omega = \{\Omega_d \mid d = 1, 2, \dots, k\}$, where Ω_d is the probability of

generating degree- d encoding symbols and $\sum_{d=1}^k \Omega_d = 1$. The k input symbols form a set $I = \{i_j \mid j = 1, 2, \dots, k\}$. Another set $C = \{c_l \mid l = 1, 2, \dots\}$ is comprised of all encoding symbols. The symbol length can be arbitrary, from a single bit to a binary vector. The degree $d(c_l)$ of encoding symbol c_l is chosen with probability $\Omega_{d(c_l)}$. Then $d(c_l)$ distinct input symbols are uniformly and randomly selected as neighbors of symbol c_l . The value of symbol c_l is the exclusive-or of its $d(c_l)$ neighbors. The $(1 + \epsilon)k$ received encoding form a set C_r , where $C_r \subseteq C$. Robust Soliton distribution (RSD) is introduced in [9] and summarized in Definition 2. Table I shows more notations and their definitions.

Definition 2: RSD is a probability set $\Omega^r = \{\Omega_d^r \mid d = 1, 2, \dots, k\}$, where

$$y \equiv c \cdot \ln \left(\frac{k}{\delta} \right) \sqrt{k}$$

$$\zeta(d) = \begin{cases} \frac{y}{kd} & \text{for } d = 1, 2, \dots, \frac{k-y}{y} \\ \frac{y}{k} \ln \left(\frac{y}{\delta} \right) & \text{for } d = \frac{k}{y} \\ 0 & \text{for } d > \frac{k}{y} \end{cases}$$

$$\Omega_d^r = \frac{\zeta(d) + \Omega_d^s}{\sum_{d=1}^k \zeta(d) + \Omega_d^s}$$

and $\Omega^s = \{\Omega_d^s \mid d = 1, 2, \dots, k\}$ is the ideal Soliton distribution with $\Omega_1^s = 1/k$ and $\Omega_d^s = 1/(d(d-1))$ for $d = 2, 3, \dots, k$. The parameters c and δ are for adjustment.

B. Drawbacks of LT Code With Short Code Length

As mentioned in [18], BP decoding process is more likely to terminate when the average number of degree-1 encoding symbols

$$R(\rho) = (1 + \epsilon)(k - \rho) \left(\Omega^r \left(\frac{\rho}{k} \right) + \frac{1}{1 + \epsilon} \ln \frac{k - \rho}{k} \right) + O(1)$$

$$= \left(\sum_{d=1}^k \Omega_d f(\rho, d) + (k - \rho) \ln \frac{k - \rho}{k} \right) + O(1) \quad (1)$$

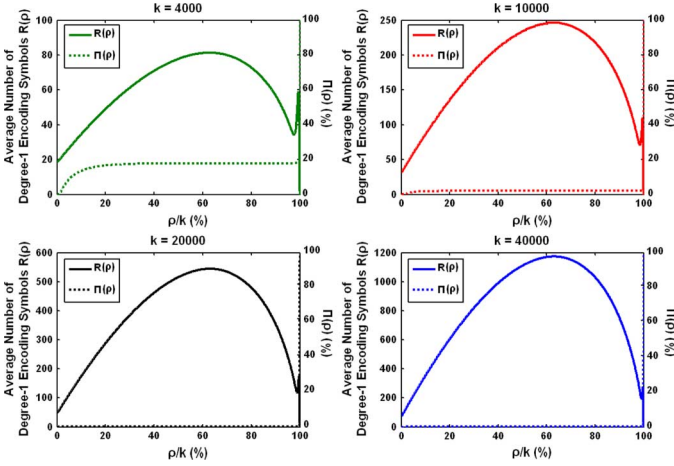


Fig. 1. Average number of degree-1 encoding symbols $R(\rho)$ and cumulative probability of decoding termination $\Pi(\rho)$ versus the ratio of decoded input symbols ρ/k using RSD of $c = 0.02$ and $\delta = 0.01$ for different code lengths.

is relatively small, where ρ is the number of decoded input symbols, $f(\rho, d) = (1 + \epsilon)(k - \rho)d(\frac{\rho}{k})^{d-1}$ and distribution Ω is used for encoding. A BP decoder is in a state with $\rho = 0$ when $(1 + \epsilon)k$ encoding symbols are collected but the decoding process has not started yet. The result in (1) is a good approximation based on the assumption that encoding symbols are allowed to choose their neighbors with replacement during encoding. For most RSD, the average number of degree-1 encoding symbols is relatively small around $\rho = 0$, leading to early decoding termination. Moreover, with fixed RSD parameters c and δ , the average number of degree-1 encoding symbols becomes proportional to k . As k decreases, the problem of early decoding termination becomes more serious.

Fig. 1 shows the average number of degree-1 encoding symbols versus the ratio of decoded input symbols ρ/k . The results are for different code lengths with $\epsilon = 0.1$ and RSD of $c = 0.02$ and $\delta = 0.01$. The corresponding cumulative probability of decoding termination based on one million simulations is also depicted. The curves of the cumulative probability with $k = 20000$ and $k = 40000$ are close to the x-axes. It can be seen that the average number of degree-1 encoding symbols is relatively low around $\rho/k = 0\%$, where decoding termination occurs frequently. As k decreases, more decoding processes terminate when $\rho/k \leq 20\%$.

III. PROPOSED ENCODING SCHEME AND DEGREE DISTRIBUTION MODIFICATION

A. Increasing Degree-1 Proportion and Its Problems

The average number of degree-1 encoding symbols is equal to $(1 + \epsilon)k\Omega_1$ when $\rho = 0$. Given ϵ and k , this number increases with the degree-1 proportion. According to (1), any change in a degree distribution only affects the term $\Omega_d f(\rho, d)$. Besides,

$$\frac{f(\rho, 1)}{f(\rho, d)} = \frac{1}{d \left(\frac{\rho}{k}\right)^{d-1}} \geq 1,$$

for $d > 1$ and $0 < \rho \leq \frac{k}{2}$. Thus we increase the degree-1 proportion to obtain larger average number of degree-1 encoding

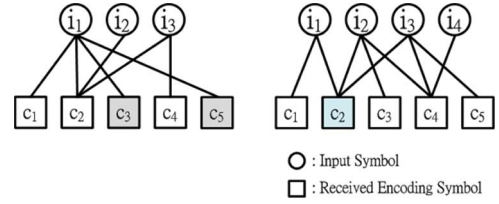


Fig. 2. Examples to show repeated degree-1 symbols (c_3 and c_5 on the left) and a redundant symbol (c_2 on the right).

symbols for $0 \leq \rho \leq \frac{k}{2}$. Decoding termination can then be relieved within this range.

However, with increased degree-1 proportion, a receiver collects more encoding symbols carrying redundant information. If two received degree-1 encoding symbols connect to the same input symbol, we define the one with larger index as a repeated degree-1 symbol. A more general definition is given in the following, where $\Lambda(c_i)$ denotes a set of input symbols connected by encoding symbol c_i .

Definition 3: Received encoding symbol c_i is a repeated degree-1 symbol if $d(c_i) = 1$ and $\exists c_j \in C_r$, for $j < i$, such that $\Lambda(c_i) = \Lambda(c_j)$.

On the left of Fig. 2, input symbol i_1 is connected by received degree-1 encoding symbols c_1 , c_3 and c_5 . According to Definition 3, c_3 and c_5 are repeated degree-1 symbols.

In the following, we define a received degree- d encoding symbol, $d > 1$, as a redundant symbol if its neighbors are fully connected by received degree-1 encoding symbols.

Definition 4: Received encoding symbol $c_j \in C_r$, $d(c_j) > 1$, is a redundant symbol if $\Lambda(c_j) \in Y$, where

$$Y = \{i_l \mid \exists c_x \in (\Lambda(i_l) \cap C_r) \text{ and } d(c_x) = 1\}$$

and $\Lambda(i_l)$ is a set of encoding symbols connecting to input symbol i_l .

On the right of Fig. 2, c_2 is a redundant symbol because all of its neighbors i_1 , i_2 and i_3 are connected by received degree-1 encoding symbols c_1 , c_3 and c_5 , respectively. Input symbols i_1 , i_2 and i_3 remain decodable after removing redundant symbol c_2 .

When g input symbols are connected by received degree-1 encoding symbols, a received degree- d encoding symbol has probability $\binom{g}{d} / \binom{k}{d}$ to be redundant, where $g \geq d > 1$. This probability decreases as d becomes larger. So redundant symbols can be reduced by decreasing the proportion of low-degree (except for degree-1) encoding symbols. We introduce this method to modifying RSD later.

B. Non-Repetitive Encoding Scheme

Non-Repetitive (NR) encoding scheme is proposed to avoid generating repeated degree-1 symbols. An NR encoder connect the first k degree-1 encoding symbols to different input symbols. Meanwhile, encoding symbols of degree larger than one are generated in the same way as LT encoding.

Algorithm 1 illustrates NR encoding scheme. A sequence $U = \{u_z = z \mid z = 1, 2, \dots, k\}$ is produced in advance. This sequence records whether an input symbol is already connected by degree-1 encoding symbols. Pointer γ is initially set

to k , pointing at the end of sequence U . If the degree of encoding symbol c_j is great than one, $d(c_j) > 1$, $d(c_j)$ input symbols are uniformly and randomly selected as the neighbors of symbol c_j . If $d(c_j) = 1$, a number t is uniformly and randomly chosen between 1 and γ , and input symbol u_t is selected as the neighbor of symbol c_j . Then elements u_t and u_γ in sequence U are swapped, followed by number γ being reduced by one. Number γ is reset to k when it decreases to zero. This mechanism ensures that neighbors of the first k degree-1 encoding symbols never repeat. Each input symbol has equal probability to be connected by degree-1 encoding symbols.

Algorithm 1 Non-Repetitive Encoding Scheme

```

1: for  $z = 1$  to  $k$  do
2:    $u_z \leftarrow z$ 
3: end for
4:  $\gamma \leftarrow k$ 
5: for  $j = 1$  to  $\infty$  do
6:   Choose degree  $d(c_j)$  based on  $\Omega$ 
7:   if  $d(c_j) = 1$  then
8:     Uniformly and randomly select number  $t$  from 1 to  $\gamma$ 
9:      $c_j \leftarrow u_t$ 
10:     $u_t \leftrightarrow u_\gamma$ 
11:     $\gamma \leftarrow \gamma - 1$ 
12:   if  $\gamma = 0$  then
13:      $\gamma \leftarrow k$ 
14:   end if
15: else
16:   Uniformly and randomly select  $d(c_j)$  different numbers
      $y_1^{c_j}, y_2^{c_j}, \dots, y_{d(c_j)}^{c_j}$  from 1 to  $k$ 
17:    $c_j \leftarrow i_{y_1^{c_j}} \oplus i_{y_2^{c_j}} \oplus \dots \oplus i_{y_{d(c_j)}^{c_j}}$ 
18: end if
19: Transmit encoding symbol  $c_j$ 
20: end for

```

NR encoding scheme is different from systematic code. Typically, the degree of the first k encoding symbols is equal to one for systematic code. Then as channel erasure probability increases, a receiver collects lower proportion of degree-1 encoding symbols, and vice versa. For NR encoding scheme, the degree of each encoding symbol is randomly decided. The first k degree-1 encoding symbols scatter randomly in an encoding symbol stream and they connect to different input symbols. On average, the proportion of received degree-1 encoding symbols stays unchanged despite different channel erasure probability.

C. Upper Bound of Degree-1 Proportion

Without loss of generality, an NR encoder generates encoding symbols using degree distribution Ω . A receiver collects encoding symbols sequentially. Each encoding symbol experiences a channel erasure probability P_e during transmission. Assume that among $(1 + \epsilon)k$ received encoding symbols, symbol c_j has the largest index, where $j \geq (1 + \epsilon)k$. In other words, the received symbols are from a set of encoding symbols $c_1, c_2, \dots, c_j$. As mentioned before, the first k degree-1 encoding symbols connect to different input symbols. If the

number of degree-1 encoding symbols among $c_1, c_2, \dots, c_j$ is no greater than k , this receiver is free from collecting repeated degree-1 symbols.

Let us denote the probability mass function of binomial distribution by

$$B(P, x_1, x_2) = \binom{x_2}{x_1} P^{x_1} (1 - P)^{x_2 - x_1}.$$

Then the probability that symbol c_j is collected and $j - (1 + \epsilon)k$ out of encoding symbols $c_1, c_2, \dots, c_{j-1}$ are erased during transmission is

$$\phi(j) = (1 - P_e)B(P_e, j - (1 + \epsilon)k, j - 1),$$

where $(1 - P_e)$ is the probability of collecting encoding symbol c_j . The probability that there exist no more than k degree-1 encoding symbols among symbols $c_1, c_2, \dots, c_j$ is

$$\theta(j) = \sum_{q=0}^k B(\Omega_1, q, j). \quad (2)$$

Then the probability of collecting repeated degree-1 symbols is no greater than $\sum_{j \geq (1+\epsilon)k} (1 - \theta(j))\phi(j)$.

Our objective now is to ensure that the probability of receiving repeated degree-1 symbols is lower than parameter $\hat{Q}$. We consider the worst channel condition with an erasure probability $\hat{P}_e$. Then given $k, \epsilon, \hat{P}_e$ and $\hat{Q}$, a corresponding $\hat{\Omega}_1$ can be derived numerically such that

$$\hat{Q} = \sum_{j \geq (1+\epsilon)k} (1 - \theta(j))\phi(j). \quad (3)$$

By setting $\Omega \leq \hat{\Omega}_1$, the probability of collecting repeated degree-1 symbols is guaranteed to be lower than $\hat{Q}$ for $0\% \leq P_e \leq \hat{P}_e$. The upper bound $\hat{\Omega}_1$ of the degree-1 proportion decreases as $\hat{P}_e$ grows.

Let us denote by M , the average number of transmitted encoding symbols, where $M \geq (1 + \epsilon)k$. On average, there are $M\Omega_1$ degree-1 encoding symbols and $M - (1 + \epsilon)k$ encoding symbols erased during transmission. In addition, we have $P_e = \frac{M - (1+\epsilon)k}{M}$. Given k, ϵ and Ω_1 , smaller P_e corresponds to smaller $M\Omega_1$. Then less degree-1 encoding symbols lead to smaller probability of collecting repeated degree-1 symbols. When $P_e \leq \hat{P}_e$, the probability of receiving repeated degree-1 symbols is guaranteed below $\hat{Q}$. However, when $P_e > \hat{P}_e$, the probability of receiving repeated degree-1 symbols could grow over $\hat{Q}$. The performance of NR encoding scheme degrades accordingly since receivers become more likely to collect repeated degree-1 symbols.

D. Modification of RSD

The modification of RSD includes two steps that are proceeded iteratively as shown in Algorithm 2. We denote the modified distribution by $\tilde{\Omega}$. Symbol loss probability is denoted by $L_{\Omega^r}^{LT}$ and $L_{\tilde{\Omega}}^{NR}$ for LT encoding with RSD Ω^r and NR encoding scheme with distribution $\tilde{\Omega}$, respectively. In the first step, the degree-1 proportion is increased by α_1 each time, where $0 \leq \alpha_1 \leq 1$ (line 8 to line 12 in Algorithm 2). Parameter α_1 controls how fast the degree-1 proportion increases. The first step

stops when the degree-1 proportion is no smaller than $\hat{\Omega}_1$ or symbol loss probability can no longer be reduced. In the second step, the degree- d proportion is reduced by α_2 each time with the degree- $(d+1)$ proportion increased by α_2 (line 13 to line 19 in Algorithm 2), where $0 \leq \alpha_2 \leq 1$. This step starts with $d = 2$ until $d = k - 1$ or symbol loss probability can no longer be reduced. The proportion of low-degree (except for degree-1) encoding symbols decreases in the second step. Thus the probability of receiving redundant symbols can be reduced as mentioned in the end of Subsection III-A.

Algorithm 2 Robust Soliton Distribution Modification

```

1:  $\tau \leftarrow 0$ 
2: for  $d = 2$  to  $k$  do
3:    $\lambda_d \leftarrow 0$ 
4: end for
5:  $T_1 \leftarrow L_{\Omega^r}^{LT}$ 
6:  $T_2 \leftarrow L_{\Omega^r}^{LT}$ 
7: while  $T_2 \leq T_1$  do
8:   while  $\Omega_1 < \tilde{\Omega}_1$  and  $L_{\Omega}^{NR} \leq T_1$  do
9:      $T_1 \leftarrow L_{\Omega}^{NR}$ 
10:     $\tau \leftarrow \tau + \alpha_1$ 
11:    Update  $L_{\Omega}^{NR}$ 
12:   end while
13:   for  $d = 2$  to  $k - 1$  do
14:     while  $\tilde{\Omega}_d \geq \alpha_2$  and  $L_{\Omega}^{NR} \leq T_2$  do
15:        $T_2 \leftarrow L_{\Omega}^{NR}$ 
16:        $\lambda_d \leftarrow \lambda_d + \alpha_2$ 
17:       Update  $L_{\Omega}^{NR}$ 
18:     end while
19:   end for
20: end while

```

Let us denote the increase in the degree-1 proportion after the modification of RSD by τ . The amount moved from the degree- d proportion to the degree- $(d+1)$ proportion is denoted by λ_d . Then the relationship between RSD Ω^r and modified distribution $\tilde{\Omega}$ can be given by

$$\tilde{\Omega}_1 = \Omega_1^r + \tau, \quad (4)$$

$$\tilde{\Omega}_2 = \begin{cases} 0 & \text{if } \Omega_2^r \leq \tau + \lambda_2 \\ \Omega_2^r - \tau - \lambda_2 & \text{else} \end{cases}, \quad (5)$$

and for $d = 3, 4, \dots, k$, see (6) at the bottom of the page, where $0 \leq \tilde{\Omega}_d \leq 1$ and $\sum_{d=1}^k \tilde{\Omega}_d = 1$.

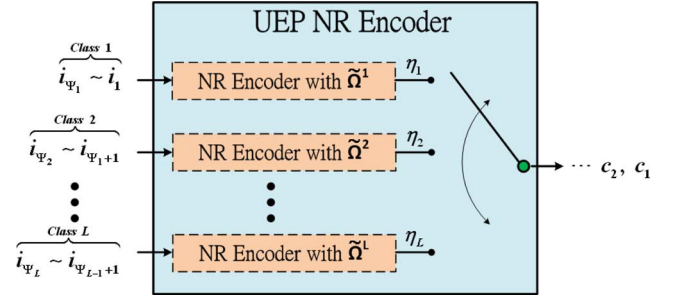


Fig. 3. Integration of NR encoders to implement UEP.

RSD can also be modified with respect to successful decoding probability by replacing $L_{\Omega^r}^{LT}$ with $S_{\Omega^r}^{LT}$ and L_{Ω}^{NR} with S_{Ω}^{NR} in Algorithm 2, where $S_{\Omega^r}^{LT}$ and S_{Ω}^{NR} denote the successful decoding probability of LT encoding and NR encoding scheme, respectively. In this paper, we run 10000 simulations to measure probabilities L_{Ω}^{NR} , $L_{\Omega^r}^{LT}$, S_{Ω}^{NR} and $S_{\Omega^r}^{LT}$ during the modification process.

IV. PROPOSED UEP SCHEME

The k input symbols are partitioned into L classes. Class l is more important than class j if $l < j$. The j -th class contains k_j input symbols and $k_1 + k_2 + \dots + k_L = k$. For indexing simplicity, we further introduce $\Psi_j = \sum_{q=1}^j k_q$ and $\Psi_0 = 0$. Then input symbols $i_{\Psi_{j-1}+1}, i_{\Psi_{j-1}+2}, \dots, i_{\Psi_j}$ belong to class j for $j = 1, 2, \dots, L$.

NR encoders are integrated to implement UEP as depicted in Fig. 3. Using distribution $\tilde{\Omega}^j$, the j -th NR encoder generates encoding symbols from the input symbols of class j . An encoding symbol from the j -th NR encoder has probability η_j to be transmitted each time, where $\sum_{j=1}^L \eta_j = 1$. The average overhead of the j -th class is denoted by $\bar{\epsilon}_j$. Given $(1 + \epsilon)k$ received encoding symbols, on average, we have

$$\bar{\epsilon}_j = \frac{(1 + \epsilon)k\eta_j - k_j}{k_j}.$$

Overhead $\bar{\epsilon}_j$ and number k_j are used in designing distribution $\tilde{\Omega}^j$.

The UEP ability is achieved by setting $\bar{\epsilon}_l \geq \bar{\epsilon}_j$, where class l is more important than class j . For example, we now have input symbols belonging to two classes. Given that the number of received encoding symbols is fixed, the overhead of class one $\bar{\epsilon}_1$ can be increased by reducing the overhead of class two $\bar{\epsilon}_2$. It is worth noting that $\bar{\epsilon}_1$ increases nearly k_2/k_1 times faster than

$$\tilde{\Omega}_d = \begin{cases} 0 & \text{if } \sum_{j=2}^d \Omega_j^r < \tau + \lambda_d \\ \sum_{j=2}^d \Omega_j^r - \tau - \lambda_d & \text{if } \sum_{j=2}^d \Omega_j^r < \tau \text{ and } \sum_{j=2}^d \Omega_j^r \geq \tau + \lambda_d \\ \Omega_d^r + \lambda_{d-1} - \lambda_d & \text{else} \end{cases} \quad (6)$$

$\bar{\epsilon}_2$ decreases. When k_2 is much larger than k_1 , we can greatly increase the overhead of class one with a small decrease in the overhead of class two.

V. SIMULATION RESULTS

NR encoding scheme is first compared with LT encoding, sliding-window (SW) encoding [16] and duplicate-symbols (DS) encoding [17]. We then compare the proposed UEP scheme with the UEP schemes of [14], [15] and [17], using settings identical to those used in [17]. In the followings, we consider the worst channel condition with an erasure probability $\hat{P}_e = 50\%$. The upper bound $\hat{\Omega}_1$ of the degree-1 proportion is derived using parameters $\epsilon = 0.1$ and $\hat{Q} = 0.01$. Each data point is the result averaged over one million simulations. The PSNR performance is determined between the original and reconstructed videos. In the application of these UEP schemes, input symbols are partitioned into two classes, and referred to as the block of most important bits (MIB) and the block of least important bits (LIB) in [17]. Thus we have k_{MIB} , k_{LIB} , $\hat{\Omega}^{MIB}$, $\hat{\Omega}^{LIB}$, $\bar{\epsilon}_{MIB}$, $\bar{\epsilon}_{LIB}$, η_{MIB} and η_{LIB} stand for k_1 , k_2 , $\hat{\Omega}_1$, $\hat{\Omega}_2$, $\bar{\epsilon}_1$, $\bar{\epsilon}_2$, η_1 and η_2 , respectively, and $\eta_{LIB} = 1 - \eta_{MIB}$.

A. Comparisons Between Various Encoding Schemes

Using $k = 2000$, NR encoding scheme is compared with LT encoding, DS and SW with a window size of 2000 input symbols. DS employs an *expanding factor* EF such that its degree distribution is designed using code length $EF \times k$. A buffer capable of storing 20000 encoding symbols is used for SW during BP decoding process. All of the encoding schemes are implemented without pre-codes, such as low-density parity-check (LDPC) code.

Fig. 4 shows symbol loss probability as a function of overhead for LT encoding and NR encoding scheme. Corresponding successful decoding probability versus overhead is depicted in Fig. 5. RSD of $c = 0.02$ and $\delta = 0.01$ is used for LT encoding and further modified into distribution $\hat{\Omega}^b$ for NR encoding scheme as listed in Table V. The main advantage of LT encoding, SW and DS over NR encoding scheme is the fact that their performance is channel-independent. For $0\% \leq P_e < 100\%$, the performance of LT encoding, SW and DS stays unchanged because the degree of encoding symbols and their connections are randomly decided. Thus channel erasure probability need not be taken into account in code design [9], [16]. The performance of NR encoding scheme stays unchanged only for $0\% \leq P_e \leq \hat{P}_e$. This can be observed from Figs. 4 and 5, where the curves of $P_e \leq 50\%$ overlap with each other (the upper bound of the degree-1 proportion $\hat{\Omega}_1$ is derived with $\hat{P}_e = 50\%$). When $P_e = 70\%$, the probability of receiving repeated degree-1 symbols is no longer guaranteed lower than $\hat{Q}$. Receivers collect more repeated degree-1 symbols, leading to the performance degradation in terms of both symbol loss probability and successful decoding probability. As P_e grows to 99%, the performance of NR encoding scheme degrades further.

Fig. 6 compares the symbol loss probability of LT encoding, SW, DS and NR encoding scheme. RSD of $c = 0.1$ and $\delta = 0.5$ is used for LT encoding, SW and DS (same as in [17]). This RSD is further modified into distribution $\hat{\Omega}^a$ for NR encoding scheme as shown in Table V. The curves of SW decline rapidly when

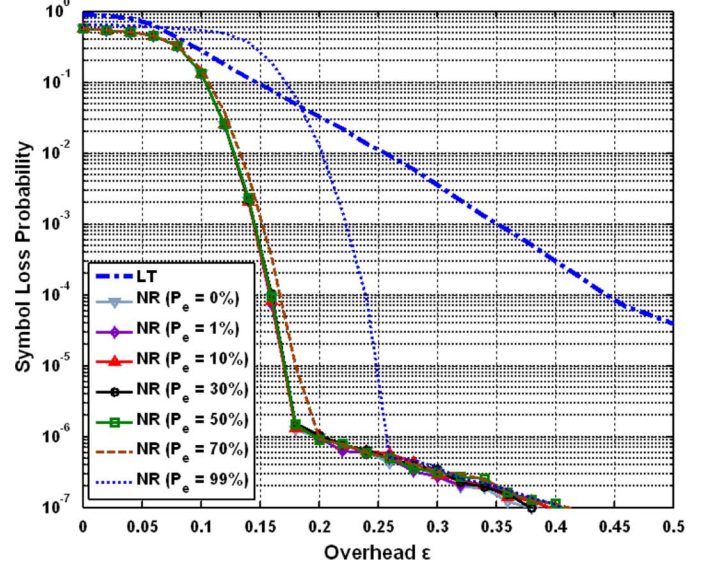


Fig. 4. Symbol loss probability versus overhead ϵ with $k = 2000$. RSD of $c = 0.02$ and $\delta = 0.01$ is used for LT encoding and further modified into distribution $\hat{\Omega}^b$ for NR encoding scheme. Encoding symbols experience different channel erasure probability P_e during transmission.

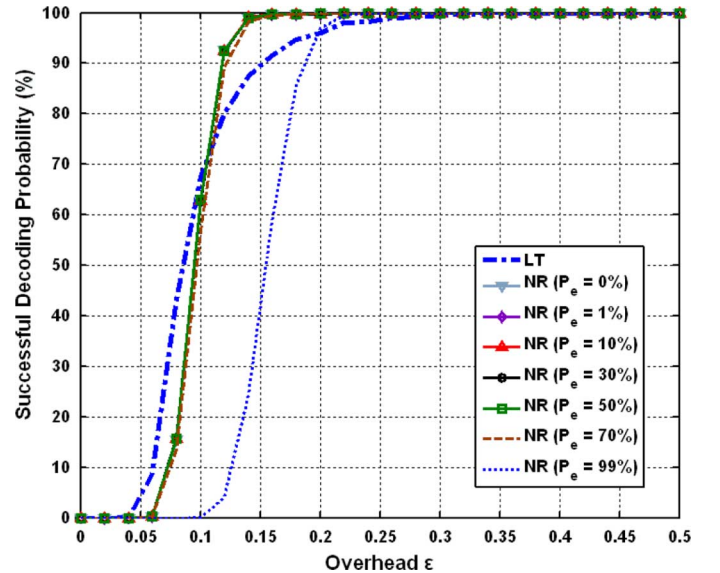


Fig. 5. Successful decoding probability versus overhead ϵ with $k = 2000$. RSD of $c = 0.02$ and $\delta = 0.01$ is used for LT encoding and further modified into distribution $\hat{\Omega}^b$ for NR encoding scheme. Encoding symbols experience different channel erasure probability P_e during transmission.

overhead is small, which is similar to the results shown in Fig. 5 in [16]. It can be seen in Fig. 6 that both SW and DS outperform NR encoding scheme in terms of symbol loss probability for $0 \leq \epsilon \leq 0.16$. Additionally, DS achieves lower symbol loss probability than NR encoding scheme when overhead is larger than 0.33. Nevertheless, compared to other encoding schemes with RSD of $c = 0.02$ and $\delta = 0.01$, the symbol loss probability of NR encoding scheme with distribution $\hat{\Omega}^b$ declines more rapidly for $0.1 \leq \epsilon \leq 0.18$ as shown in Fig. 7.

Cumulative probability of decoding termination versus ratio ρ/k is depicted in Fig. 8 with different overhead. On average, SW performs better in reducing the probability of decoding

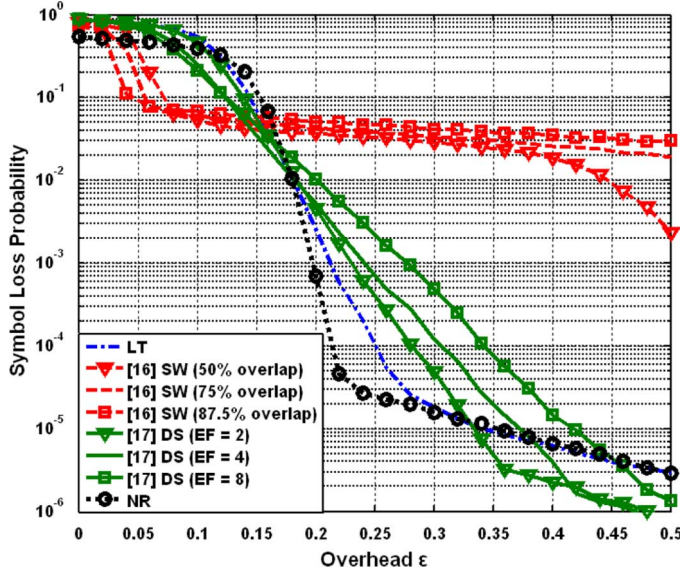


Fig. 6. Symbol loss probability versus overhead with $k = 2000$. SW has a window size of 2000 input symbols and it uses a buffer capable of storing 20000 encoding symbols during BP decoding process. RSD of $c = 0.1$ and $\delta = 0.5$ is used for LT encoding, SW and DS, which is modified into distribution $\tilde{\Omega}^a$ for NR encoding scheme as listed in Table V. Channel erasure probability is 50%.

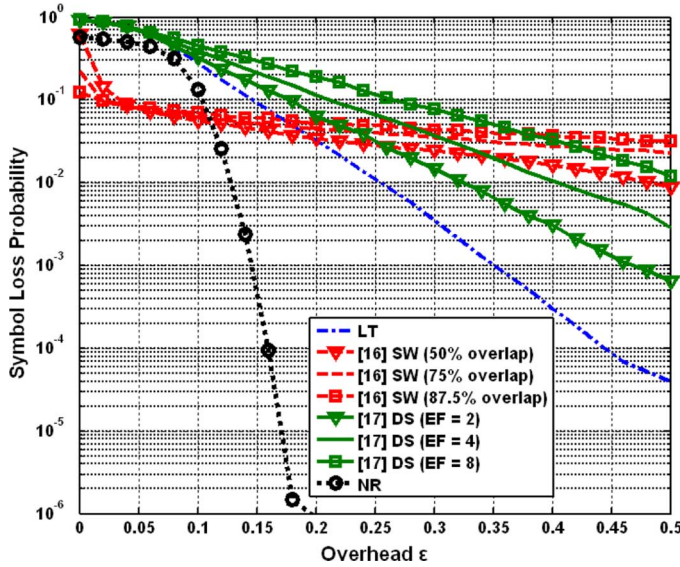


Fig. 7. Symbol loss probability versus overhead with $k = 2000$. SW has a window size of 2000 input symbols and it uses a buffer capable of storing 20000 encoding symbols during BP decoding process. RSD of $c = 0.02$ and $\delta = 0.01$ is used for LT encoding, SW and DS, which is modified into distribution $\tilde{\Omega}^b$ for NR encoding scheme as listed in Table V. Channel erasure probability is 50%.

termination. For NR encoding scheme, decoding termination rarely occurs for $0\% \leq \rho/k \leq 40\%$ due to the increased proportion of degree-1 encoding symbols. With distribution $\tilde{\Omega}^a$, most decoding processes terminate within the range $40\% \leq \rho/k \leq 80\%$, and this situation does not improve as the overhead grows. With distribution $\tilde{\Omega}^b$, the probability of decoding termination is greatly reduced by NR encoding scheme when $\epsilon \geq 0.12$. According to Table I, the successful decoding probability is equal to $1 - \Pi(1999)$ for $k = 2000$. Then the lower the cumulative probability $\Pi(\rho)$ of decoding termination is at

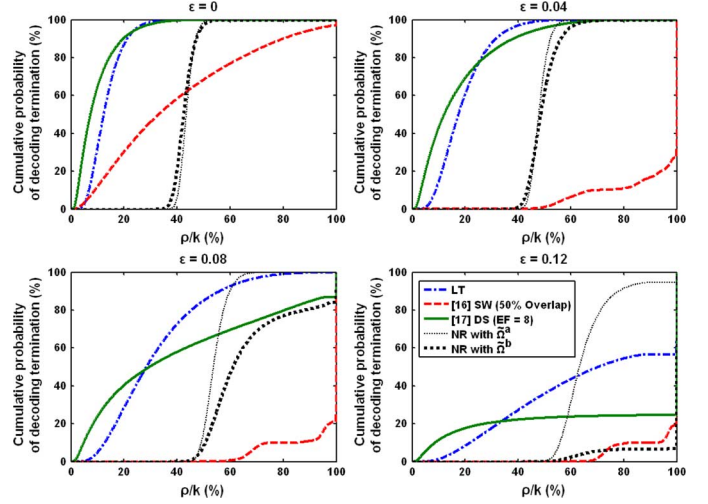


Fig. 8. Cumulative probability $\Pi(\rho)$ of decoding termination versus ratio ρ/k with $k = 2000$. SW has a window size of 2000 input symbols and it uses a buffer capable of storing 20000 encoding symbols during BP decoding process. RSD of $c = 0.1$ and $\delta = 0.5$ is used for LT encoding, SW and DS. Distributions $\tilde{\Omega}^a$ and $\tilde{\Omega}^b$ are used for NR encoding scheme as listed in Table V. Channel erasure probability is 50%.

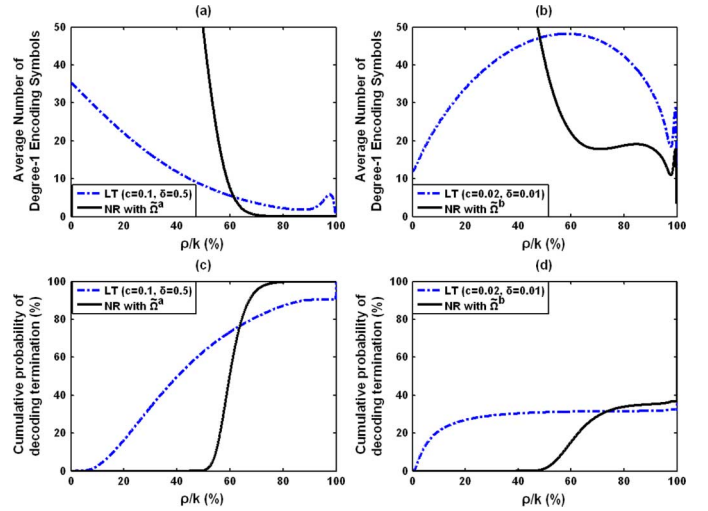


Fig. 9. Average number of degree-1 encoding symbols and cumulative probability of decoding termination versus ratio ρ/k with $k = 2000$. (a), (c) RSD of $c = 0.1$ and $\delta = 0.5$ is used for LT encoding and further modified into distributions $\tilde{\Omega}^a$ for NR encoding scheme. (b), (d) RSD of $c = 0.02$ and $\delta = 0.01$ is used for LT encoding and further modified into distributions $\tilde{\Omega}^b$ for NR encoding scheme as listed in Table V. Channel erasure probability is 50%.

the points of $\rho/k = 99.95\%$ in Fig. 8, the higher the successful decoding probability will be. As ϵ grows to 0.12, NR encoding scheme achieves higher successful decoding probability than that of other encoding schemes. This indicates that NR encoding scheme outperforms other encoding schemes in terms of successful decoding probability when overhead is sufficiently large.

Fig. 9 shows the average number of degree-1 encoding symbols versus ratio ρ/k with $\epsilon = 0.1$ for LT encoding and NR encoding scheme. Corresponding cumulative probability of decoding termination is also represented. When NR encoding scheme is applied with distributions $\tilde{\Omega}^a$ and $\tilde{\Omega}^b$, the average number of degree-1 encoding symbols is increased to 705.8

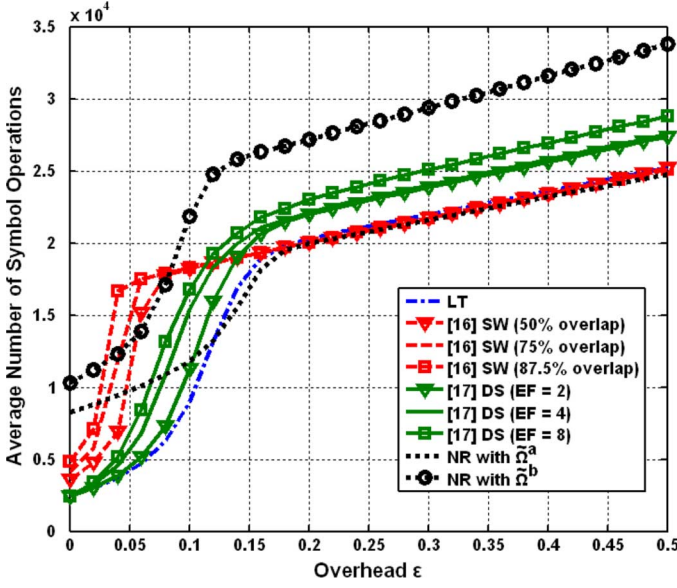


Fig. 10. Average number of symbol operations to decode 2000 input symbols versus overhead ϵ . LT encoding, DS and NR encoding scheme have a code length $k = 2000$. SW has a window size of 2000 input symbols and it uses a buffer capable of storing 20000 encoding symbols during BP decoding process. RSD of $c = 0.1$ and $\delta = 0.5$ is used for LT encoding, SW and DS. NR encoding scheme employs distributions $\tilde{\Omega}^a$ and $\tilde{\Omega}^b$. Channel erasure probability is 50%.

and 591.4, respectively, at $\rho/k = 0\%$. The results shown in Fig. 9 explain the reason why NR encoding scheme exhibits better performance in Fig. 7 than in Fig. 6. For LT encoding with RSD of $c = 0.1$ and $\delta = 0.5$, the average number of degree-1 encoding symbols is relatively low within the range $60\% \leq \rho/k \leq 90\%$, which can be seen in Fig. 9(a). Increasing degree-1 proportion leads to higher average number of degree-1 encoding symbols within the range $0\% \leq \rho/k \leq 50\%$. However, this number decreases to zero around $\rho/k = 70\%$ where most decoding processes terminate. On the contrary, for LT encoding with RSD of $c = 0.02$ and $\delta = 0.01$, the average number of degree-1 encoding symbols is relatively low around $\rho/k = 0\%$ as shown in Fig. 9(b). Accordingly, most decoding termination occurs during early stage. This problem can be relieved by NR encoding scheme as Fig. 9(d) shows. The results indicate that NR encoding scheme is able to relieve decoding termination for $0\% \leq \rho/k \leq 50\%$ but not for $50\% \leq \rho/k \leq 100\%$.

As proposed by M. Luby in [9], the run time of the encoder and decoder is measured in terms of symbol operations. A symbol operation is either an exclusive-or between two symbols or a copy of one symbol to another. The average number Δ of symbol operations to generate an encoding symbol is listed in Table II. The average number of symbol operations to decode 2000 input symbols is also recorded and shown in Fig. 10 as a function of overhead. DS virtually extends the code length by EF times to design a degree distribution. So its average degree per encoding symbol is proportional to EF . More symbol operations are conducted during encoding and decoding process as parameter EF grows. NR encoding scheme with distribution $\tilde{\Omega}^b$ greatly reduces symbol loss probability as shown in Fig. 7 but at the cost of high encoding and decoding

TABLE II
AVERAGE NUMBER Δ OF SYMBOL OPERATIONS TO GENERATE AN ENCODING SYMBOL WITH $k = 2000$. RSD OF $c = 0.1$ AND $\delta = 0.5$ IS USED FOR LT ENCODING, SW AND DS. NR ENCODING SCHEME EMPLOYS DISTRIBUTIONS $\tilde{\Omega}^a$ AND $\tilde{\Omega}^b$ AS LISTED IN TABLE V. ENCODING COMPLEXITY INCREASES WITH Δ

Encoding Scheme		Δ
SW	LT	8.47
	50% overlap	8.47
	75% overlap	8.47
	87.5% overlap	8.47
DS	EF = 2	9.34
	EF = 4	10.17
	EF = 8	10.97
NR	$\tilde{\Omega}^a$	8.4
	$\tilde{\Omega}^b$	11.4

TABLE III
SVC ENCODING RESULT OF 10-MINUTE VIDEO SEQUENCE “STEFAN_NEW.YUV”. THIS VIDEO IS A CONCATENATION OF 200 COPIES OF VIDEO SEQUENCE “STEFAN.YUV”. THE FRAME SIZE IS 352×288 AND THE FRAME RATE IS 30 FPS. THE SVC REFERENCE SOFTWARE (JSVM) IS USED FOR SOURCE ENCODING WITH ONE BASE LAYER (BL) AND 10 ENHANCEMENT LAYERS (EL). THIS TABLE SHOWS THE NUMBER OF SYMBOLS, THE BIT RATE IN KBPS AND THE Y-PSNR IN DB

Decoded Layers	Size	Bitrate	PSNR
BL	400	328.5	29.2
BL + 1EL	743	610.4	31.0
BL + 2EL	902	741	32.0
BL + 3EL	1068	877.7	33.1
BL + 4EL	1619	1329.8	34.4
BL + 10EL	3800	3121	40.0

complexity. This is because distribution $\tilde{\Omega}^b$ has higher average degree per encoding symbol, compared to those used by other encoding schemes.

B. Experiment Flow for PSNR Comparisons

On the transmitter side, the SVC reference software (JSVM) is introduced as the source encoder. First, we generate 200 copies of video sequence “Stefan.yuv”, which contains 90 frames and has a frame size of 352×288 and a frame rate of 30 fps. These copies are concatenated into a new video sequence “Stefan_new.yuv” with 18,000 frames or equivalently 10 minutes in length. Second, the Fixed QP Encoder tool is employed to find the basis quantization parameters for the new video sequence. The ratio between bit rates of the base layer and all layers is set to 400:3800, same as in [17]. Then video sequence “Stefan_new.yuv” is encoded into network abstraction layer units (NALU) with one base layer (BL) and 10 enhancement layers (EL) in the quality (SNR) scalability mode. Table III shows sizes, bit rates and Y-PSNR of different layers. Group of Pictures (GOP) size is set to 16. As in [17], NALUs are divided into input symbols of 50 bytes to be encoded by different UEP schemes.

On the receiver side, first, NALUs are rebuilt from BP decoding output. An NALU is contained in multiple input symbols. Any loss of the containing input symbols leads to the drop of this NALU. Second, nalufilter, which is developed in [19], is applied to filter out undecodable NALUs according to their dependency. Third, the SVC decoder reconstructs video sequence “Stefan_rec.yuv” from the remaining NALUs. We discard an

entire GOP if the coded slice of an instantaneous decoder refresh (IDR) picture of this GOP is lost. Any lost frame is re-filled with the temporarily closest frame by the framefiller of [19]. PSNR comparison is performed between video sequences “Stefan_new.yuv” and “Stefan_rec.yuv”.

For SVC, the enhancement layer can be partially decoded. Part of this layer remains decodable despite losses of network abstraction layer units (NALUs). Then the decodability of the enhancement layer is related to the symbol loss probability of the LIB block. On the other hand, the base layer is decodable only when the entire NALUs of this layer are recovered. Any loss of the NALUs will lead to the drop of the entire base layer. Thus the decodability of the base layer depends on the successful decoding probability of the MIB block. Moreover, when an MIB block is undecoded, the corresponding LIB block must be discarded. According to Table III, the decoding of the entire MIB block contributes 29.2 to the PSNR. The decoding of the entire LIB block is able to increase the PSNR by 10.8 (40 – 29.2). All in all, the successful decoding probability of the MIB block influences the PSNR more than the symbol loss probability of the LIB block.

C. Settings of UEP Schemes for PSNR Comparisons

Input symbols are partitioned such that the MIB block and the LIB block comprise one base layer and 10 enhancement layers, respectively. According to Table III, the MIB block contains 400 input symbols and the LIB block contains 3400 input symbols. We have $k_{MIB} = 400$ and $k_{LIB} = 3400$. However, in practice, the ratio between the input symbol number of the two layers varies over time. So the MIB block is allowed to contain a small portion of input symbols belonging to the enhancement layer.

The UEP encoding scheme of [14] is tested with both the fixed degree distribution (2) as in [14] and RSD of $c = 0.1$ and 0.5 . The UEP encoding scheme of [15] is tested under two settings. First, RSD of $c = 0.03$ and $\delta = 0.5$ is used for encoding the MIB block and the fixed distribution for encoding the LIB block as in [17]. Second, RSD of $c = 0.03$ and $\delta = 0.5$ is used for encoding the MIB block but RSD of $c = 0.1$ and $\delta = 0.5$ is used for encoding the LIB block. For the UEP scheme of [17], RSD of $c = 0.1$ and $\delta = 0.5$ is used with parameters $RF = 2$ and $EF = 12$ as in [17].

We modify RSD of $c = 0.01$ and $\delta = 0.02$ into distribution $\tilde{\Omega}^{MIB}$ with respect to the successful decoding probability for encoding the MIB block. RSD of $c = 0.03$ and $\delta = 0.02$ is modified into distribution $\tilde{\Omega}^{LIB}$ with respect to the symbol loss probability for encoding the LIB block. These two distributions are listed in Table VI.

D. PSNR Comparisons

Fig. 11 is a plot of the PSNR versus the overhead ϵ for the proposed UEP scheme. Channel erasure probability is 50%. This figure shows the influence of different η_{MIB} on the PSNR performance. It can be seen that the PSNR at $\epsilon = 0$ increases with η_{MIB} . This is because more encoding symbols are generated to protect base layer as η_{MIB} grows. However, higher η_{MIB} leads to less protection for the enhancement layer. So the curve of $\eta_{MIB} = 0.129$ rises the slowest within the range

TABLE IV
AVERAGE NUMBER Δ OF SYMBOL OPERATIONS TO GENERATE AN ENCODING SYMBOL WITH $k = 3800$ FOR DIFFERENT UEP SCHEMES. ENCODING COMPLEXITY INCREASES WITH Δ

Encoding Scheme		Δ
[14]	RSD	9.23
	Fixed Degree	5.85
[15]	RSD	9.07
	Fixed Degree	6.04
[17]	RSD	12.14
	Proposed	13.4

$0.06 \leq \epsilon \leq 0.12$ where the enhancement layer begins to contribute to the PSNR. In the following simulations, η_{MIB} is set to 0.125.

Fig. 14 is a plot of the PSNR versus overhead ϵ with different P_e for the proposed UEP scheme. Fig. 12 represents the corresponding successful decoding probability of the MIB block, and the symbol loss probability of the LIB block is shown in Fig. 13. These three figures represent the influence of channel erasure probability on the performance of the proposed UEP scheme. The curves of $P_e = 0\%$, 1% , 10% , 30% and 50% overlap with each other in these figures. The overlap demonstrates that the performance of our UEP scheme stays unchanged for $0\% \leq P_e \leq 50\%$. The PSNR performance degrades as P_e increases to 70% since the probability of receiving repeated degree-1 symbols is no longer guaranteed lower than $\bar{Q}$. When $P_e = 99\%$, the PSNR performance degrades further.

Compared to LT encoding, SW and DS, NR encoding scheme requires smaller overhead to achieve low symbol loss probability. For example of $k = 2000$, using distribution Ω^b , NR encoding scheme achieves a symbol loss probability of 10^{-4} at $\epsilon = 0.16$ as shown in Fig. 7. Other encoding schemes require an overhead of at least 0.25 to achieve the same probability as shown in Figs. 6 and 7. In comparison, the former saves an overhead of 0.09. This allows us to decrease the overhead of the LIB block while maintaining its symbol loss probability comparable to that of other UEP schemes. Meanwhile, the overhead of the MIB block can be increased to ensure its high successful decoding probability.

By setting $\eta_{MIB} = 0.125$, we protect the MIB block with large overhead in the application of our UEP scheme. Let us take $\epsilon = 0.05$ as an example. We have the overhead of the MIB block $\bar{\epsilon}_{MIB} = \frac{(1+\epsilon)k\eta_{MIB}-k_{MIB}}{k_{MIB}} = \frac{1.05 \times 3800 \times 0.125 - 400}{400} = 0.247$, which is sufficiently large to achieve high successful decoding probability. Besides, $\bar{\epsilon}_{LIB} = \frac{(1+\epsilon)k\eta_{LIB}-k_{LIB}}{k_{LIB}} = \frac{1.05 \times 3800 \times 0.875 - 3400}{3400} = 0.027$. The LIB block yields an overhead of 0.023 ($0.05 - 0.027$) in exchange for a great increase of 0.197 ($0.247 - 0.05$) in the overhead of the MIB block.

Fig. 15 shows the PSNR versus overhead ϵ for the proposed UEP scheme and the UEP schemes of [14], [15] and [17]. Channel erasure probability is 50%. Fig. 16 shows the corresponding successful decoding probability of the MIB block, and the symbol loss probability of the LIB block is represented in Fig. 17. For our UEP scheme, the PSNR is mainly contributed by the MIB block within the range $0 \leq \epsilon \leq 0.08$ where the symbol loss probability of the LIB block is still high (above 0.279). When $\epsilon > 0.08$, the symbol loss probability of the LIB block declines rapidly and this block begins to contribute to

TABLE V
MODIFIED RSDs $\tilde{\Omega}^a$ AND $\tilde{\Omega}^b$ WITH $k = 2000$

Modified Robust Soliton Distributions							
$\tilde{\Omega}^a$				$\tilde{\Omega}^b$			
i	$\tilde{\Omega}_i^a$	i	$\tilde{\Omega}_i^a$	i	$\tilde{\Omega}_i^b$	i	$\tilde{\Omega}_i^b$
1	0.3208	20	0.003	1	0.2688	23	0.0021
2	0.0001	21 ~ 22	0.0029	2	0.0001	24	0.002
3	0.28	23	0.0025	3	0.3728	25	0.0019
4	0.0869	24	0.0022	4	0.0808	26	0.0017
5	0.0477	25 ~ 26	0.0021	5	0.048	27	0.0015
6	0.0319	27	0.0019	6	0.0324	28	0.0015
7	0.0227	28 ~ 30	0.0017	7	0.024	29	0.0015
8	0.0173	31	0.0015	8	0.017	30	0.0013
9	0.0145	32	0.0014	9	0.0143	31	0.0011
10	0.0114	33 ~ 35	0.0012	10	0.0109	32 ~ 33	0.001
11	0.0093	36 ~ 37	0.0011	11	0.0093	34 ~ 36	0.0009
12	0.0076	38 ~ 41	0.001	12	0.0077	37 ~ 39	0.0008
13	0.0075	42	0.0009	13	0.0067	30 ~ 42	0.0007
14	0.006	43 ~ 46	0.0008	14	0.0054	43 ~ 47	0.0006
15	0.0051	47 ~ 49	0.0007	15	0.0053	48 ~ 54	0.0005
16	0.0046	50 ~ 52	0.0006	16	0.0042	55 ~ 61	0.0004
17 ~ 18	0.004	53	0.0005	17	0.0041	62 ~ 67	0.0003
19	0.0034	54	0.0689	18	0.0031	68 ~ 70	0.0002
				19	0.003	71 ~ 112	0.0001
				20	0.0028	113 ~ 183	≈ 0.0
				21	0.0026	184	0.0351
				22	0.0024		

TABLE VI
MODIFIED RSDs $\tilde{\Omega}^{MIB}$ AND $\tilde{\Omega}^{LIB}$ WITH $k_{MIB} = 400$ AND $k_{LIB} = 3400$

Modified Robust Soliton Distributions							
$\tilde{\Omega}^{MIB}$				$\tilde{\Omega}^{LIB}$			
i	$\tilde{\Omega}_i^{MIB}$	i	$\tilde{\Omega}_i^{MIB}$	i	$\tilde{\Omega}_i^{LIB}$	i	$\tilde{\Omega}_i^{LIB}$
1	0.2774	17	0.0039	1	0.2802	22	0.0022
2	0.0	18	0.0035	2	0.0	23	0.002
3	0.3991	19	0.0031	3	0.3886	24	0.0019
4	0.0845	20	0.0028	4	0.0836	25	0.0017
5	0.0509	21	0.0026	5	0.0502	26	0.0016
6	0.0341	22	0.0024	6	0.0335	27	0.0015
7	0.0244	23	0.0022	7	0.024	28	0.0014
8	0.0184	24	0.002	8	0.018	29	0.0013
9	0.0144	25	0.0018	9	0.014	30	0.0012
10	0.0115	26	0.0017	10	0.0112	31	0.0011
11	0.0095	27	0.0016	11	0.0092	32 ~ 33	0.001
12	0.0079	28	0.0015	12	0.0077	34 ~ 35	0.0009
13	0.0067	29	0.0014	13	0.0065	36 ~ 37	0.0008
14	0.0058	30 ~ 244	≈ 0.0	14	0.0056	38 ~ 40	0.0007
15	0.005	245	0.0153	15	0.0048	41 ~ 44	0.0006
16	0.0044			16	0.0042	45 ~ 48	0.0005
				17	0.0037	49 ~ 55	0.0004
				18	0.0033	56 ~ 66	0.0003
				19	0.003	67 ~ 86	0.0002
				20	0.0027	87 ~ 835	≈ 0.0
				21	0.0024	836	0.0059

the PSNR. In comparison, the UEP scheme of [17] and our scheme have similar PSNR performance for $0.08 \leq \epsilon \leq 0.5$. Meanwhile, the latter outperforms the former in terms of the PSNR for $0 \leq \epsilon < 0.08$. This is because even when the overall overhead ϵ is small, our scheme protects the MIB block with large overhead $\bar{\epsilon}_{MIB}$ to ensure its high successful decoding probability. It is worth noting that for the UEP scheme of [17], the successful decoding probability of the MIB block can be increased by using larger parameter RF . However, after increasing RF from 2 to 3, the symbol loss probability of the LIB block rises at the same time, leading to a decrease of the overall PSNR as mentioned in [17]. On the other hand, our scheme

always outperforms the UEP schemes of [14] and [15] in terms of the successful decoding probability of the MIB block. The symbol loss probability of the LIB block of our scheme drops lower than that of other UEP schemes when $\epsilon \geq 0.12$. As a result, our scheme achieves better PSNR performance than the UEP schemes of [14] and [15].

The average number Δ of symbol operations to generate an encoding symbol is listed in Table IV for different UEP schemes. In addition, the average number of symbol operations during BP decoding process is recorded and depicted in Fig. 18 as a function of overhead. The curves of [14] and [15] with fixed degree distribution (dash lines) and the curve of [17] (solid line

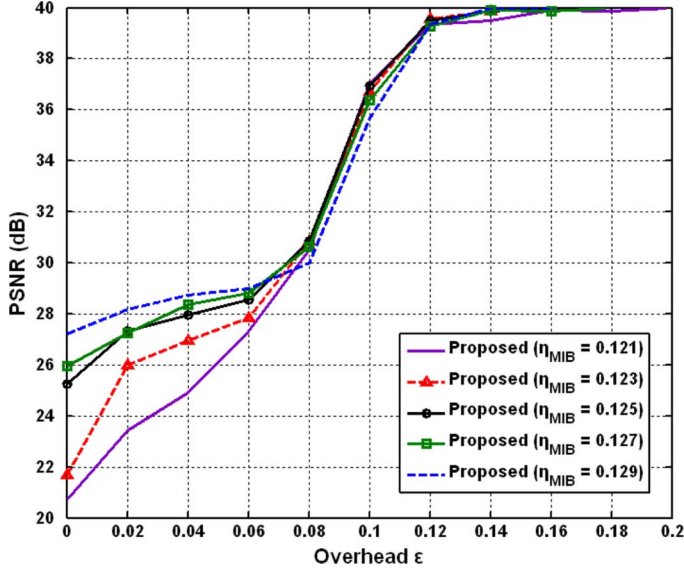


Fig. 11. PSNR versus overhead with $k_{MIB} = 400$ and $k_{LIB} = 3400$ and different η_{MIB} for the proposed UEP scheme. Channel erasure probability is 50%.

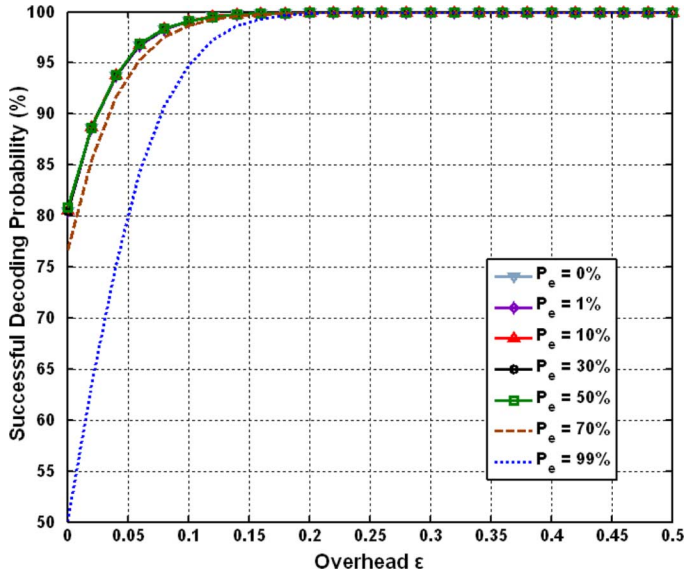


Fig. 12. Successful decoding probability of the MIB block versus overhead with $k_{MIB} = 400$ for the proposed UEP scheme. Encoding symbols experience different channel erasure probability P_e during transmission.

with square marks) exhibit similar trend as that of the decoding time shown in Fig. 15 in the reference paper [17]. Compared to other UEP schemes, our scheme employs distributions with higher average degree per encoding symbol, leading to more symbol operations during encoding and decoding processes.

We also measure the performance of the UEP schemes when the number of transmitted encoding symbols is fixed. For each code block, 4560 ($1.2k$) encoding symbols are transmitted through channels with different erasure probability. The PSNR performance of the proposed UEP scheme and the UEP schemes of [14], [15] and [17] is depicted in Fig. 19. The corresponding successful decoding probability of the MIB block and the symbol loss probability of the LIB block are shown in Figs. 20 and 21, respectively. Let us first compare

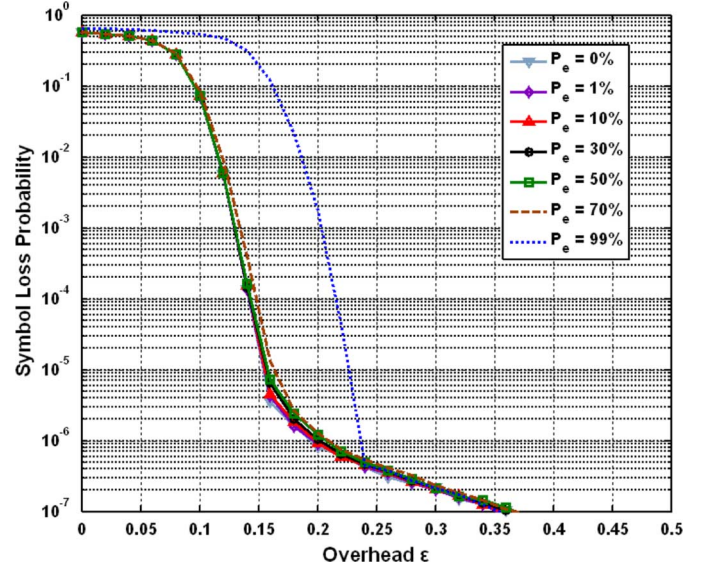


Fig. 13. Symbol loss probability of the LIB block versus overhead with $k_{LIB} = 3400$ for the proposed UEP scheme. Encoding symbols experience different channel erasure probability P_e during transmission.

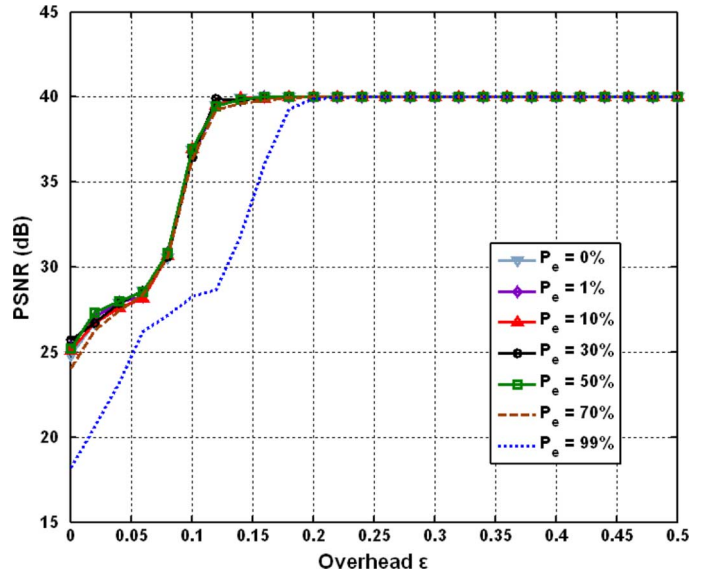


Fig. 14. PSNR versus overhead with $k_{MIB} = 400$, $k_{LIB} = 3400$ for the proposed UEP scheme. Encoding symbols experience different channel erasure probability P_e during transmission.

the performance of the UEP scheme of [17] and our UEP scheme. When $P_e \leq 6\%$, the PSNR performance loss of both UEP schemes is insignificant. As P_e grows to 8%, the successful decoding probability of the MIB block of our UEP scheme is 3.6% higher than that of the UEP scheme of [17], but the latter achieves lower symbol loss probability. As a joint result, the PSNR value of these two schemes are very close when $P_e = 8\%$. The difference of the successful decoding probability of the MIB block between these two UEP schemes is 31% when $P_e = 10\%$. This difference is large enough that our UEP scheme achieves higher PSNR than the UEP scheme of [17] although the latter has lower symbol loss probability of the LIB block. For the same reason, the PSNR of our UEP scheme decreases slower than that of the UEP scheme of [17]

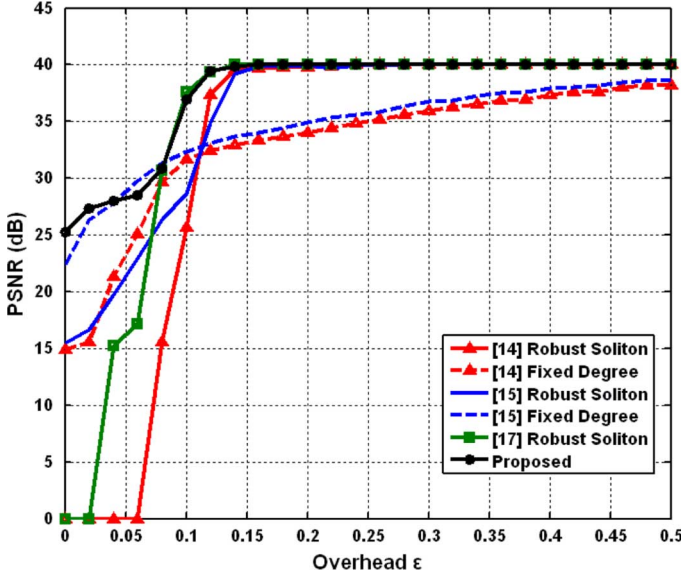


Fig. 15. PSNR versus overhead with $k_{MIB} = 400$ and $k_{LIB} = 3400$ for different UEP schemes. Channel erasure probability is 50%.

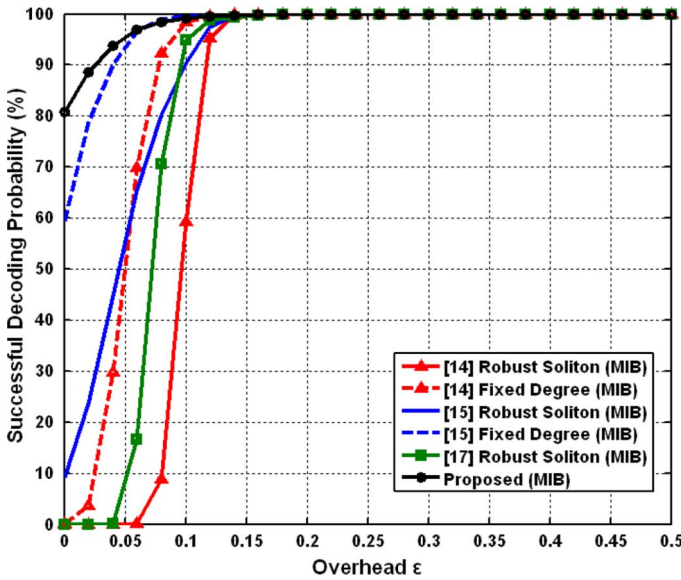


Fig. 16. Successful decoding probability of the MIB block versus overhead with $k_{MIB} = 400$ for different UEP schemes. Channel erasure probability is 50%.

as P_e grows over 10%. Following similar observation, it is conceivable that our UEP scheme outperforms other UEP schemes in terms of the PSNR. When the PSNR is smaller than 20, it means most of the video frames are lost. Once a frame is decoded, it is repeatedly used to replace lost frames, leading to pauses on the screen up to several minutes. Especially, when the PSNR is around 15, only one or two of the entire 18000 frames are decoded. Without decoded frames, the PSNR drops to zero rapidly, indicating the loss of the whole video.

VI. CONCLUSION

In this paper, our objective is to improve the performance of short-length LT codes on channels with limited erasure

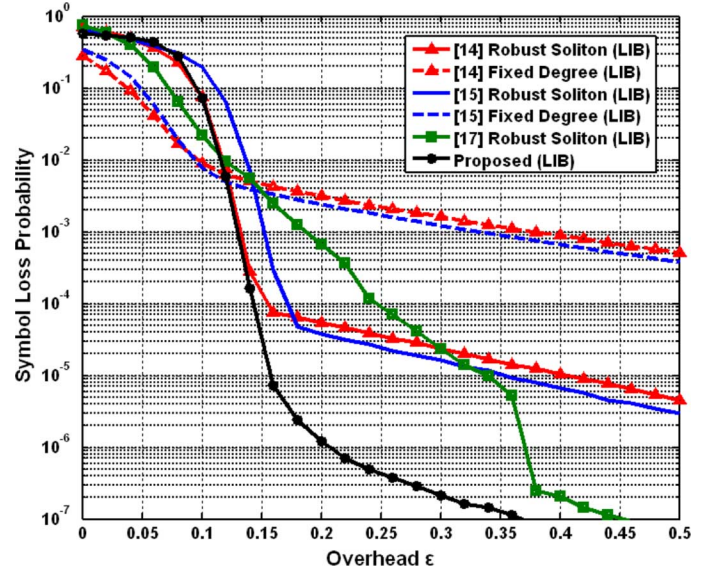


Fig. 17. Symbol loss probability of the LIB block versus overhead with $k_{LIB} = 3400$ for different UEP schemes. Channel erasure probability is 50%.

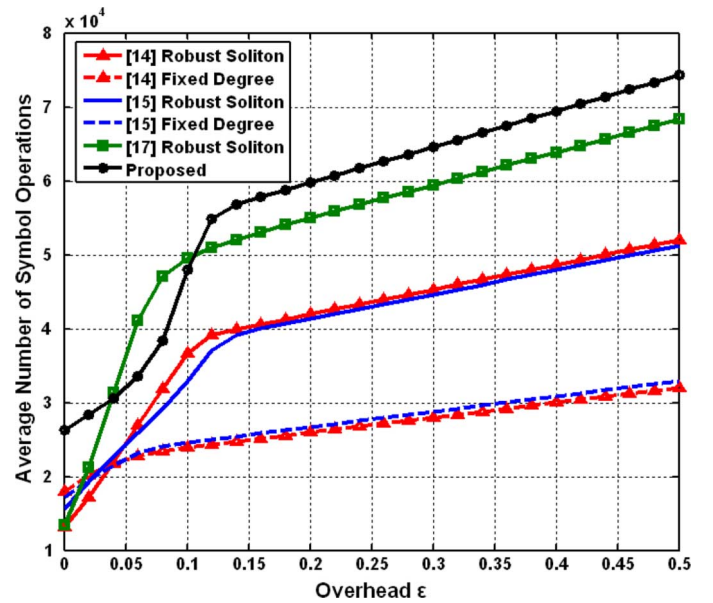


Fig. 18. Average number of symbol operations to decode 3800 input symbols versus overhead with $k_{MIB} = 400$ and $k_{LIB} = 3400$ for different UEP schemes. Channel erasure probability is 50%.

probability ($P_e \leq 50\%$). We first increase the degree-1 proportion in RSD, and propose NR encoding scheme to generate more non-repetitive degree-1 encoding symbols to relieve early decoding termination. The low-degree proportion in RSD is also decreased to avoid generating redundant symbols. For $0\% \leq P_e \leq 50\%$, NR encoding scheme requires smaller overhead than LT encoding, SW and DS to achieve low symbol loss probability. For example of $k = 2000$, NR encoding scheme achieves a symbol loss probability of 10^{-4} at $\epsilon = 0.16$, while LT encoding, SW and DS require an overhead of at least 0.25 to achieve the same probability. However, compared to the latter, NR encoding scheme employs degree distributions with

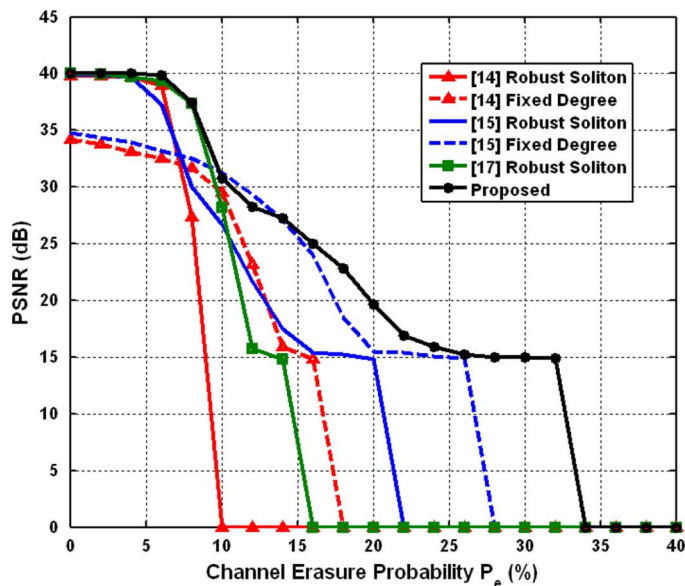


Fig. 19. PSNR versus channel erasure probability P_e with $k_{MIB} = 400$ and $k_{LIB} = 3400$ for different UEP schemes. The number of transmitted encoding symbols is 4560 (1.2k) per code block.

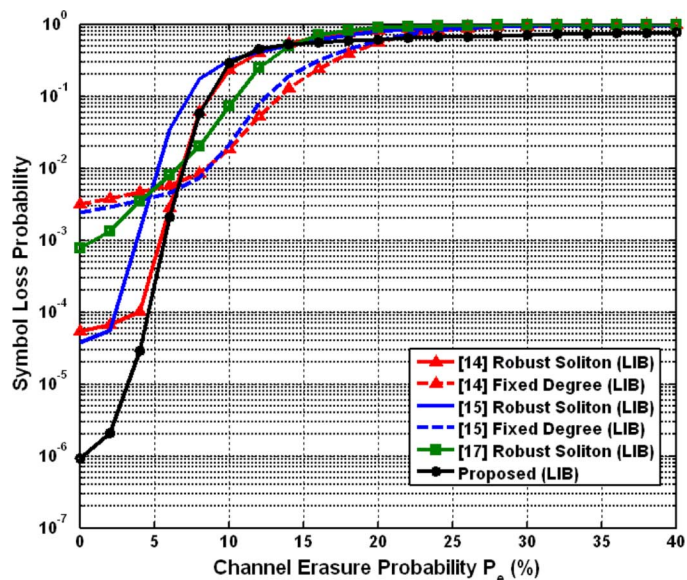


Fig. 21. Symbol loss probability of the LIB block versus channel erasure probability P_e with $k_{LIB} = 3400$ for different UEP schemes. The number of transmitted encoding symbols is 4560 (1.2k) per code block.

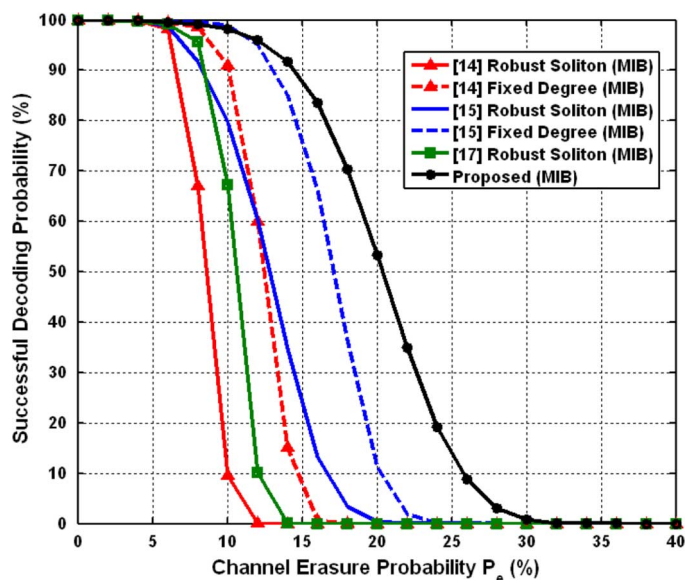


Fig. 20. Successful decoding probability of the MIB block versus channel erasure probability P_e with $k_{MIB} = 400$ for different UEP schemes. The number of transmitted encoding symbols is 4560 (1.2k) per code block.

higher average degree per encoding symbol, resulting in higher encoding and decoding complexity.

Second, we integrate two NR encoders to implement UEP. The overhead of the MIB block and the LIB block can be different. For SVC, the MIB block contributes more to the PSNR than the LIB block. As mentioned above, compared to LT encoding and DS, NR encoding scheme requires smaller overhead to achieve low symbol loss probability. This allows us to decrease the overhead of the LIB block while maintaining its symbol loss probability comparable to that of other UEP schemes. Meanwhile, we are able to increase the overhead of the MIB block to ensure its high successful decoding probability. As a result, our UEP scheme outperforms the UEP schemes of

[14], [15] and [17] in terms of the PSNR for $0\% \leq P_e \leq 50\%$, but at the cost of higher encoding and decoding complexity. The performance of our UEP scheme begins to degrade when $P_e > 50\%$ because receivers become more likely to collect repeated degree-1 symbols. Users can choose between our UEP scheme and the other schemes according to P_e . The former could be more desirable for $0\% \leq P_e \leq 50\%$, while the latter perform better when $P_e > 50\%$.

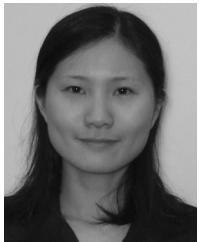
REFERENCES

- [1] H. Wang and C.-C. J. Kuo, "Robust video multicast with joint network coding and AL-FEC," in *Proc. IEEE Int. Symp. Circuits and Systems (ISCAS 2008)*, 2008, pp. 2062–2065.
- [2] D. Gomez-Barquero, D. Gozalvez, and N. Cardona, "Application layer FEC for mobile TV delivery in IP datacast over DVB-H systems," *IEEE Trans. Broadcast.*, vol. 55, no. 2, pp. 396–406, 2009.
- [3] I. de Foz, F. Fraile, R. Belda, and J. C. Guerri, "Analysis and evaluation of adaptive LDPC AL-FEC codes for content download services," *IEEE Trans. Multimedia*, vol. 14, no. 3, pp. 641–650, 2012.
- [4] L. Qian, D. L. Jones, K. Ramchandran, and S. Appadwedula, "A general joint source-channel matching method for wireless video transmission," in *Proc. Data Compression Conf.*, 1999, pp. 414–424.
- [5] X. Jaspas, C. Guillemot, and L. Vandendorpe, "Joint source-channel turbo techniques for discrete-valued sources: From theory to practice," *Proc. IEEE*, pp. 1345–1361, 2007.
- [6] N. Raja, Z. Xiong, and M. Fossorier, "Combined source-channel coding of images under power and bandwidth constraints," *EURASIP J. Adv. Signal Process.*, 2007.
- [7] J. W. Byers, M. Luby, M. Mitzenmacher, and A. Rege, "A digital fountain approach to reliable distribution of bulk data," in *Proc. ACM SIGCOMM*, 1998, pp. 56–67.
- [8] D. J. C. MacKay, "Fountain codes," *IEE Proc. Commun.*, vol. 152, pp. 1062–1068, Dec. 2005.
- [9] M. Luby, "LT codes," in *Proc. 43rd Annu. IEEE Symp. Foundations of Computer Science*, 2002, pp. 271–280.
- [10] M. Luby, T. Gasiba, T. Stockhammer, and M. Watson, "Reliable multimedia download delivery in cellular broadcast networks," *IEEE Trans. Broadcast.*, vol. 53, no. 1, pp. 235–246, 2007.
- [11] D. Sejdinovic, R. Piechocki, A. Doufexi, and M. Ismail, "Fountain code design for data multicast with side information," *IEEE Trans. Wireless Commun.*, vol. 8, no. 10, pp. 5155–5165, 2009.
- [12] S.-K. Lee, Y.-C. Liu, H.-L. Chiu, and Y.-C. Tsai, "Fountain codes with PAPR constraint for multicast communications," *IEEE Trans. Broadcast.*, vol. 57, no. 2, pp. 319–325, 2011.

- [13] A. Shokrollahi, "Raptor codes," *IEEE Trans. Inf. Theory*, vol. 52, no. 6, pp. 2551–2567, 2006.
- [14] 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, 2007.
- [15] D. Sejdinovic, D. Vukobratovic, A. Doufexi, V. Senk, and R. J. Piechocki, "Expanding window fountain codes for unequal error protection," in *Proc. 41st Asilomar Conf. Signals, Systems and Computers*, 2007, pp. 1020–1024.
- [16] P. Cataldi, M. Grangetto, T. Tillo, E. Magli, and G. Olmo, "Sliding-window raptor codes for efficient scalable wireless video broadcasting with unequal loss protection," *IEEE Trans. Image Process.*, vol. 19, no. 6, pp. 1491–1503, 2010.
- [17] S. Ahmad, R. Hamzaoui, and M. M. Al-Akaidi, "Unequal error protection using fountain codes with applications to video communication," *IEEE Trans. Multimedia*, vol. 13, no. 1, pp. 92–101, 2011.
- [18] G. Maatouk and A. Shokrollahi, "Analysis of the second moment of the LT decoder," in *Proc. Int. Symp. Information Theory, ISIT*, 2009, pp. 2326–2330.
- [19] A. Detti *et al.*, "SVEF: An open-source experimental evaluation framework," in *Proc. IEEE MediaWIN 2009*, Sousse, Tunisia, 2009.



Kuo-Kuang Yen received the B.S. and M.S. degrees from National Chiao-Tung University, Hsinchu, Taiwan, in 2005 and 2007, respectively, both in electronics engineering, where he is currently working toward the Ph.D. degree in electronics engineering. His general research interests include coding theory and multimedia communications.



Yen-Chin Liao received the B.S. and M.S. degrees from the Department of Communication Engineering, National Chiao Tung University, Hsinchu, Taiwan, in 2000 and 2002, respectively, and the Ph.D. Degree from National Chiao-Tung University, Hsinchu, Taiwan, in 2008, in electronics engineering.

From 2002 to 2003 she joined ISSC Technologies Corporation. From 2008 to 2010, she was with Ralink Technology Corporation and was involved in IEEE802.11n/ac projects. From 2010 to 2012 she joined MobileDevice Corporation and was engaged in baseband algorithm design for IEEE802.16e and GSM products. She is currently with Electronics Engineering Department, National Chiao-Tung University as a post doctoral researcher. Her research interests include signal processing for communication systems, coding theory, wireless communication, and VLSI signal processing.



Chih-Lung Chen received the B.S., M.S., and Ph.D. degrees from National Chiao-Tung University, Hsinchu, Taiwan, in 2004, 2006 and 2012, respectively, all in electronics engineering. His general research interests include VLSI implementation of error control codes and wireless communication systems.



John K. Zao (SM'01) received his B.A.Sc. (Hon.) degree in engineering science and M.A.Sc. degree in electrical engineering from University of Toronto and later S.M. and Ph.D. degrees in computer science from Harvard University. Dr. Zao has served as a Senior Member of Technical Staff (1994–1999) and then a Principal Member of Technical Staff (1999–2002) in the Information Security Department of BBN Technologies. While at BBN, he served as the principal investigator of several DARPA funded research projects including: Security Architecture for Global Mobile IP Support (MoIPS), Policy Based Security Management (PBSM), Policy Management for Defense Information Assurance Networks (Pledge) and Formal Information Assurance System Modeling (AVDA). He joined the Computer Science Department of National Chiao-Tung University in Taiwan in 2004. His current research interests include: H.264 SVC Heterogeneous Multicasting using Rateless UEP FEC, Pervasive Mobile Fog Computing and Distributed Brain Computer Interactions.



Hsie-Chia Chang received the B.S., M.S., and Ph.D. degrees from National Chiao Tung University, Hsinchu, Taiwan, in 1995, 1997, and 2002, respectively, all in electronics engineering department.

From 2002 to 2003, he was with OSP/DE1 in MediaTek Corporation, working in the area of decoding architectures for Combo single chip. In February 2003, he joined the faculty of the Electronics Engineering Department, National Chiao-Tung University, where he has been a Professor since August 2010. His research interests include algorithms and VLSI architectures in signal processing, especially for error control codes and crypto-systems. Recently, he has also committed himself to designing high code-rate ECC schemes for flash memory and multi-Gb/s chip implementations for wireless communications.

Dr. Chang served as the associate editor of IEEE Transactions on Circuits and Systems I: Regular papers since 2012. He also served as Technique Program Committee (TPC) member for IEEE A-SSCC 2011 and 2012. Dr. Chang was the recipient of the Outstanding Youth Electrical Engineer Award from Chinese Institute of Electrical Engineering in 2010, and the Outstanding Youth Researcher Award from Taiwan IC Design Society in 2011.