This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE ICC 2011 proceedings

Low-Density Generator Matrix Codes for IP Packet Video Streaming with Backward Compatibility

Yoshihide Tonomura†, Daisuke Shirai†, Masahiko Kitamura†, Takayuki Nakachi†, Tatsuya Fujii† and Hitoshi Kiya‡
†NTT Network Innovation Laboratories,
Nippon Telegraph and Telephone Corporation, 239-0847 JAPAN
‡Department of Information and Communication Systems Engineering,
Faculty of System Design, Tokyo Metropolitan University, 191-0065 JAPAN

***Abstract*—In this paper we propose a method of constructing packet-level LDGM codes that offer backward compatibility with conventional viewing devices. Our proposed method makes it possible to watch content even if the viewer does not support any FEC module. Moreover, the method also improves the coding efficiency through the combined use of packet division and interleaving methods. In general, there is a tradeoff between computation complexity and performance, but our proposed method improves coding efficiency by using a message passing decoding scheme that does not require any additional computation. The coding efficiency of the proposed method is evaluated both experimentally and theoretically.**

I. Introduction

With recent advances in computational power and the broadband infrastructure, the Internet-based video streaming service is receiving a lot of attention. Since raw digital image contents are too large to transmit economically, video standards have been created to compress the data. Currently, the international JPEG standard for digital still images and MPEG-2 for digital motion pictures are the favorite tools. Advanced versions such as JPEG 2000 and AVC/H.264 are also attracting great attention.

The latest broadband infrastructure and compression technology makes it easy for us to enjoy video streaming services. However, irrecoverable errors are possible in best-effort networks, where packet loss is frequent. Coded data is especially susceptible to these errors which seriously impact the decoded image quality. A common approach used to protect packet data is forward error correction (FEC) codes such as [1][2][3].

Among the FEC codes proposed to date, large block methods such as Low-Density Parity-Check (LDPC) codes[4], Low-Density Generator Matrix (LDGM) codes[5][6] and LT-codes [2] are beneficial because they offer increased correction capacity. The combination of a parity check matrix using a sparse matrix and a message-passing algorithm (MPA) offers linear-time decoding. Therefore, these codes can use a large block length (e.g. $n = 10^7$) on one block and this characteristic allows data transmission rates close to the theoretical maximum[5][7].

However, the combination of a sparse matrix and an MPA does not work well. This is primarily due to the fact that one packet is treated as one unit in packet-level FEC (PL-FC) which restricts the block length (e.g. $n = 200$) in broadband applications. Therefore, the coding efficiency is not as powerful as Reed-Solomon code[1] which achieves equality within Singleton bound.

In this paper, we propose a new scheme to improve the coding efficiency of LDGM codes for video streaming services over IP-based networks. To improve coding efficiency, we propose two methods: a packet division method and a new interleaving method that we call FEC interleaving. In the packet division method, transmitted packets, which usually have the same length as the frame size, are divided into small packet sizes. This increases the number of symbols in one block size and reduces the probability of getting hit by stopping set. However, simple packetization for IP transmission accrues burst sub-packet errors because LDGM structures have a regular cycle. To avoid these burst errors, we combine sub-packet division with interleaving. Furthermore, we apply interleaving only to FEC sub-packets. This method allows backward compatibility with conventional viewing devices and offers improved coding efficiency.

Section 2, we describe the proposed method with sub-

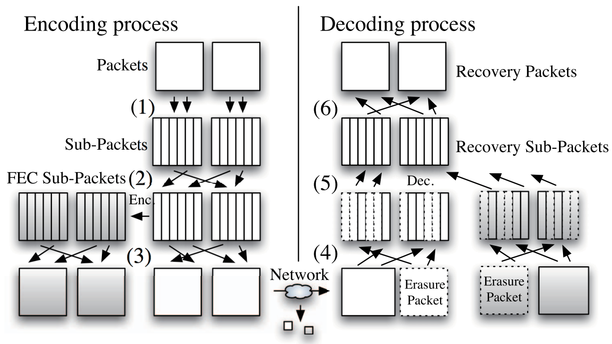

Fig. 1. Block diagram of encoding/decoding process in proposed method.

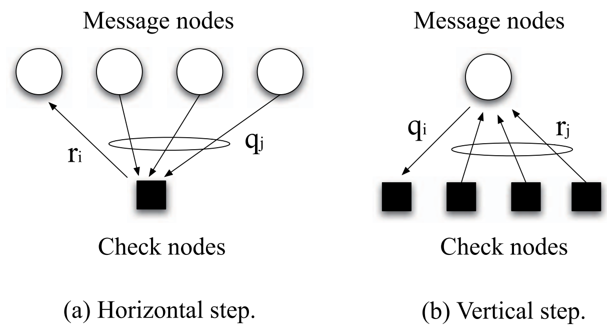

Fig. 2. Message passing rule in horizontal and vertical steps.

packet division and FEC interleaving. Section 3 evaluates the proposed LDGM scheme. Section 4 provides a theoretical analysis of the proposed method. Section 5 concludes this work by listing future directions.

II. Sub-packet Division and FEC Interleaving Methods for LDGM Codes

Fig. 1 illustrates the proposed distribution scheme for a video streaming service. The diagram shows the encode/decode process on the left/right hand side, respectively. At the encoder side, k packets are divided into $k' = k \times d$ sub-packets (Fig. 1-(1)), where d is a division number; d is 6 in the case of Fig. 1. Next, m' FEC sub-packets (i.e. $\mathbf{C}_{m'}$) are generated by the following equation after the sub-packets are interleaved (Fig. 1-(2)).

$$\mathbf{C}^t_{m'} = \left[\mathbf{T}^{-1}_{m'}\right]\left[\mathbf{P}_{m',k'}\right]\mathbf{S}^t_{k'} \pmod 2 \tag{1}$$

where $\mathbf{S}_{k'}$ represents the interleaved sub-packets which include the k' sub-packets, and t indicates a matrix transpose; $\mathbf{P}_{m',k'}$ is a sparse matrix generated using the PEG algorithm[8] and is shown as follows

$$\mathbf{P_{m',k'}} = \begin{bmatrix} & 1 & & \cdots & & \cdots & 1 & \\ 1 & & \cdots & & 1 & \cdots & & \vdots \\ \vdots & \vdots & \vdots & \vdots & \ddots & \ddots & & \\ & & 1 & & \cdots & 1 & & 1 \end{bmatrix} \tag{2}$$

$\mathbf{T}_{m'}$ is a staircase matrix as shown below.

$$\mathbf{T_{m'(staircase)}} = \begin{bmatrix} 1 & & & \\ 1 & 1 & & \\ & \ddots & \ddots & \\ & & 1 & 1 \end{bmatrix} \tag{3}$$

From equation (1), the encoding operation consists of two steps: first compute an intermediate parity vector $\mathbf{V_{m'}} = \left[\mathbf{P'_m}\right]\mathbf{S_k}$; then pass $\mathbf{V_{m'}}$ through an accumulator to create $\mathbf{C_{m'}}$. These processes are written as

$$\begin{array}{rclll} \mathbf{c_1} & = & & \sum_{j=1}^{k} P_{1j}\mathbf{s_j} \pmod 2 & \\ \mathbf{c_2} & = & \mathbf{c_1}+ & \sum_{j=1}^{k} P_{2j}\mathbf{s_j} \pmod 2 & \\ \vdots & = & \vdots & & , \\ \mathbf{c_{m'}} & = & \mathbf{c_{m'-1}}+ & \sum_{j=1}^{k} P_{mj}\mathbf{s_j} \pmod 2 & \end{array} \tag{4}$$

From equation (4), the m' parity packets $\mathbf{C_{m'}}$ can be computed in linear time.

Finally, interleaved sub-packets $\mathbf{S}_{k'}$ are packetized after de-interleaving. In contrast, the generated FEC sub-packets ($\mathbf{C}_{m'}$) are packetized after interleaving. These packets are transmitted to the decoder side (Fig.1-(3)).

At the decoder side, if packet losses are frequent, packets are divided into sub-packets and the divided sub-packets are interleaved. This process avoids burst sub-packet errors which suppresses serial sub-packet loss (Fig. 1-(4)).

Next, erasure sub-packets are recovered by the following equation which is based on the MPA algorithm (Fig. 1-(5)). The following equation is derived from Eq. (1).

$$\mathbf{0} = \left[\mathbf{P}_{m',k'}\right]\mathbf{S}^t_{k'} + \left[\mathbf{T}_{m'}\right]\mathbf{C}^t_{m'} \pmod 2 \tag{5}$$

The MPA process simply consists of using the belief propagation (BP) algorithm to solve the system of linear equations (5). BP decoding is a form of message passing and the steps for a packet erasure channel are as follows:

1) **Initialization.** At the variable nodes, variables q_i are initialized to the received packets and the missing packets are set to "error". All variables, q_i, are sent to check nodes.

2) **Horizontal step.** The horizontal step is illustrated in Fig. 1-(a). From Fig. 1-(a), the check nodes compute the values of

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE ICC 2011 proceedings

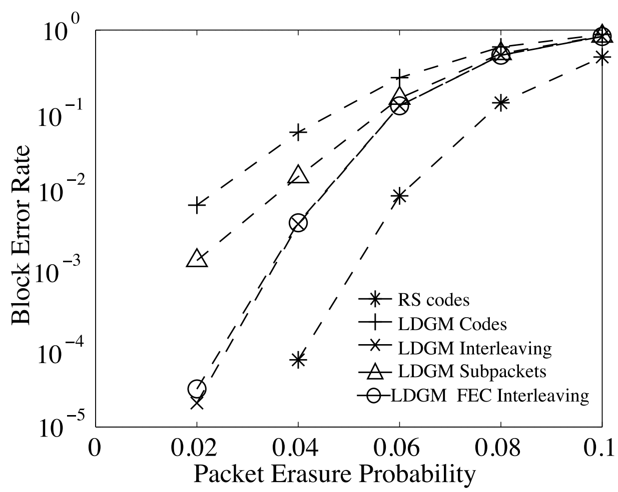

Fig. 3. The performance of each LDGM code on a packet erasure channel (dgree=3, 10%FEC).

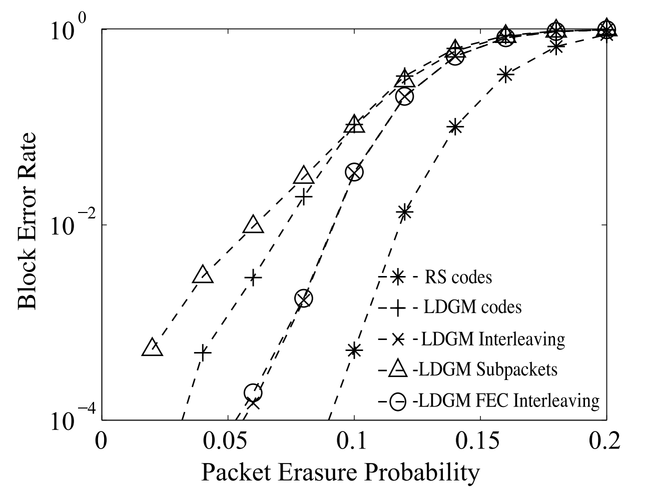

Fig. 4. The performance of each LDGM code on a packet erasure channel (dgree=3, 20%FEC).

r_i as follows:

$$r_i := \begin{cases} \sum_{j \neq i} q_j \pmod 2 & \text{if} \quad \forall j \neq i, q_i \neq error \\ error & \text{otherwise} \end{cases} \tag{6}$$

2) **Vertical step.** The vertical step is illustrated in Fig. 1-(b). From Fig. 1-(b), the variable nodes update the values of q_i as follows:

$$q_i := \begin{cases} r_j & \text{if} \quad \exists r_j, 1 \leq j \leq d_v, r_j \neq error \\ error & \text{otherwise} \end{cases} \tag{7}$$

where d_v is the maximum variable degree. The BP algorithm repeats the horizontal step to vertical step cycle.

Finally, decoded sub-packets are de-interleaved again and the decoding process is finished (Fig. 1-(6)).

For the above processes, the proposed scheme has the following advantages (we confirm the impact of these advantages in the next section):

(1) The method of sub-packet division improves coding efficiency in the case of short block lengths.
(2) The proposed scheme allows backward compatibility with conventional viewing devices that use only information packets because our scheme provides a systematic code.
(3) The FEC interleaving avoids burst errors on sub-packets and improves coding efficiency.

III. Simulation Results

We evaluated the performance of our method via computer simulation with the following experimental conditions: the communication channel is assumed to be a random packet erasure channel, the weight factor and the variable degree of nodes of the sparse matrix are set to 3, block size k is 200, FEC packet numbers, m, are 20 and 40, respectively, number of packet divisions d is 10. These conditions assume a video streaming service of about 3 M[bps]. We compared the proposed LDGM codes and RS codes with the results shown in Figs. 3, 4. In these figures, LDGM code shows normal LDGM codes without any of the proposed methods, “LDGM sub-packets” uses only the sub-packet division method without FEC interleaving, “LDGM Interleaving” combines the sub-packet division method and interleaving method (which applies to one block), and “LDGM FEC Interleaving” is the proposed scheme using packet division and FEC interleaving.

Figures 3 and 4 confirm that the proposed method improves the coding efficiency. In Fig. 4, the case of 20% FEC improves the coding efficiency by 10 - 15 [%] at the point where the block error rate is 10^{-4}; the proposed method provides the same coding efficiency as the LDGM Interleaving method. The improvement rate is evaluated by

$$\frac{\text{Rate}(\text{BER}_{proposed}) - \text{Rate}(\text{BER}_{existing})}{m/n} \times 100 \tag{8}$$

where the *Rate* function gives the achievable packet erasure probability for a given block error rate.

In Fig. 4, “LDGM sub-packet” shows low coding efficiency when compared to LDGM code. Therefore, in order to combine sub-packet division and interleaving, it is important to improve the coding efficiency for this condition. Note that, the sub-packet division method is a straightforward method and it also defines in RFC document[3], but simulation results

TABLE I
COMPARISON OF PROPOSED METHOD CHARACTERISTICS.

	Small length	Large length	Compatibility
RS codes	○	×	○
LDGM codes	×	○	○
LDGM Subpackets	△	○	×
LDGM FEC Interleaving	△	○	○

shows that the simple division ("LDMG Subpackets") does not provide good coding efficiency in the case of LDGM codes. A summary of each scheme, shown in Table I, also confirms that the proposed scheme is the only way that we can improve coding efficiency while providing backward compatibility.

IV. Theoretical Analysis of Packet Division Method

In this section, we confirm the efficiency of the proposed method through a theoretical analysis. In particular, we investigate the relationship between the number of packet divisions and coding efficiency.

In general, the finite-length analysis of LDPC codes is difficult since a huge computational load is incurred. Therefore, several methods have been proposed to analyze finite-length LDPC codes [9][10]. Ref. [9] counted out all the patterns and ref. [10] introduced a scaling function to analyze finite-length LDPC codes. However, in this paper, we evaluate the union bound of a regular LDPC ensemble under maximum-likelihood (ML) decoding for simplicity. This analysis is the same as the method that combines the sub-packet division method and long term interleaving under ML decoding.

From LDPC ensembles which are shown in Ref [9] Lemma B.2, the ensemble block error rate when using packet division under ML decoding is calculated by

$$\begin{aligned}
&\mathbb{E}_{\mathrm{LDPC}(n,x^{l-1},x^{r-1})}[\mathrm{P}_B^{ML}(G,\epsilon)] \\
&\leq \sum_{e=0}^{n}\binom{n}{e}\epsilon^{e}\bar{\epsilon}^{n-e} \\
&\cdot \min\left[1, \sum_{w=1}^{e\cdot d}\binom{e\cdot d}{w}\frac{\mathrm{coef}\left[\left(\frac{(1+y)^r+(1-y)^r}{2}\right)^{(n\cdot d)\cdot\frac{l}{r}}, y^{wl}\right]}{\binom{(n\cdot d)l}{wl}}\right] \qquad (9)
\end{aligned}$$

where ϵ is the packet erasure probability, $\bar{\epsilon}$ is the reaching probability of transmitted packets, l, r is the weight factor of the variable (check) node degree, d is the number of sub-packet divisions, n is the number of packet lengths, min[·] returns the minimum argument and coef[·] returns the coefficient of the second argument.

Eq.(9) is derived from the rank of the sparse matrix subsets which are dependent on erasure packets. The average probability that the sparse matrix subsets are not full rank matrices is given by

$$\begin{aligned}
&\Pr\{\mathrm{rank}(H_\varepsilon) < |\varepsilon|\cdot d\} \\
&= \Pr\{\exists x_\varepsilon \in \mathrm{GF}(2)^{|\varepsilon|\cdot d}\setminus\{0\} : H_\varepsilon x_\varepsilon^t = 0^t\} \\
&\leq \sum_{x\in \mathrm{GF}(2)^{|\varepsilon|\cdot d}\setminus\{0\}} \Pr\{H_\varepsilon x_\varepsilon^t = 0^t\} \\
&= \sum_{w=1}^{|\varepsilon|\cdot d}\binom{|\varepsilon|\cdot d}{w}\frac{\mathrm{coef}\left[\left(\frac{(1+y)^r+(1-y)^r}{2}\right)^{(n\cdot d)\cdot\frac{l}{r}}, y^{wl}\right]}{\binom{(n\cdot d)l}{wl}} \qquad (10)
\end{aligned}$$

Eq. (10) counts up the probability that the sparse matrix $H_\varepsilon \times x_\varepsilon^t$ equals zero. ε denotes the set of erasure probabilities and H_ε denotes a subset of matrices from H that have $|\varepsilon|$ columns. Each matrix in H_ε is made from any $|\varepsilon|$ adjacent columns in H. Similarly, x_ε denotes a subset of arbitrary codewords from x that have $|\varepsilon|$ rows. Each codeword in x_ε is made from any $|\varepsilon|$ adjacent rows in x. Please note that Eq.(10) treats edge elements as a units in a bipartite graph. Therefore, the sparse matrices used in this analysis are not the same as those used in popular configuration methods e.g. Ref. [4][8].

We evaluated the union bound of LDPC (n,l=3,r=18) codes using Eq. (9). In this simulation, the number of packets, n, was set to $240, 1200, 2400, 4800$, and the number of packet divisions, d, was set to $1, 5, 10, 20$. The result is shown in Fig. 5. Figure 5 confirms that if we use the same block length in one block the union bounds are the same in the error floor region. Therefore, the number of symbols determines the characteristics in the error floor region. For this reason, we can improve the coding efficiency even more by using the sub-packet division method with more divisions in the error floor region. Note that this analysis assumes ML decoding and it is different from the case of MPA in section II. However, we have confirmed that the proposed method does indeed improve the coding performance in ML decoding case.

Next, we evaluated the union bound to change the degree of distribution pairs. In this simulation, LDPC (n=240,l=3,r=18) codes and LDPC (n=240,l=5,r=30) codes are used and we change the number of packet divisions d. The result is shown in Fig. 6. Figure 6 shows that if we use a certain amount of density, we can improve coding efficiency under ML decoding. Moreover, our proposed method can improve the coding efficiency regardless of the density. However, ML decoding

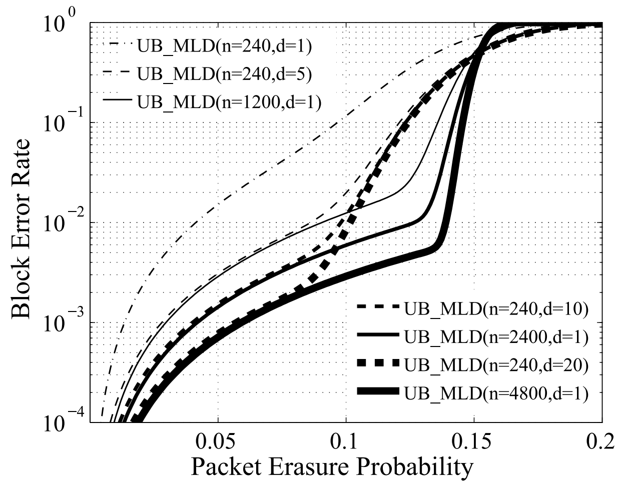

Fig. 5. Union bound on the LDPC ensembles under maximum likelihood decoding (as a function ε for d).

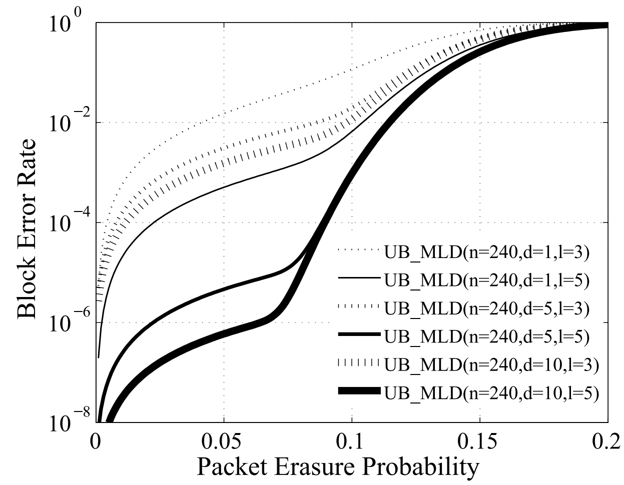

Fig. 6. Union bound on the LDPC ensembles under maximum likelihood decoding (as a function ε for l, r).

requires $O(n^3)$ arithmetical operations. The method which can reduce computational load is proposed in [11].

V. Conclusions

In this paper we proposed a method for constructing packet-level LDGM codes that allow backward compatibility with conventional viewing devices. Furthermore, we proposed a method to improve coding efficiency through the use of a combination of packet division and interleaving methods. Our proposed method improves coding efficiency in small length packet level LDGM codes. Simulations showed that our method increased the coding performance compared to existing LDGM codes. We also analyzed our method theoretically. In this analysis, we evaluated the union bound of the LDGM ensemble under ML decoding and showed that the number of symbols determines the coding performance in the error floor region. A theoretical analysis of our method combined with MPA is a future task.

References

[1] Luigi Rizzo. Effective erasure codes for reliable computer communication protocols. *Comput. Commun. Rev.*, 27(2):24–36, April 1997.

[2] M. Luby. Lt codes. In *Foundations of Computer Science, 2002. Proceedings. The 43rd Annual IEEE Symposium*, pages 271–280, November 2002.

[3] V. Roca, C. Neumann, and D. Furodet. Low density parity check (ldpc) staircase and triangle forward error correction (fec) schemes. *IETF RMT Working Group, RFC 5170*, June 2008.

[4] R.G.Gallager. *Low density parity check codes*. MIT Press, 1963.

[5] David J. C. MacKay. Good error-correcting codes based on very sparse matrices. *IEEE Trans. Inform. Theory*, 45(2):399–431, 1999.

[6] T.J. Richardson and R.L. Urbanke. Efficient encoding fo low-density parity check codes. *IEEE Transactions on Information Theory*, 47(6):638–656, February 2001.

[7] D.J.C.Mackay and R.M.Neal. Near shannon imit performance of low density parity check codes. *IEE Electron Lett.*, 32(18):1645–1646, August 1996.

[8] Xiao-Yu Hu, E. Eleftheriou, and D.M. Arnold. Regular and irregular progressive edge-growth tanner graphs. *IEEE Trans. Inform. Theory*, 51(1):386–398, January 2005.

[9] C. Di, D. Proietti, I.E. Telatar, and T.J. Richardson. Finite-length analysis of low-density parity-check codes on thebinary erasure channel. *IEEE Trans. Inform. Theory*, 48(6):1570–1579, June 2002.

[10] A. Amraoui, A. Montanari, T.J. Richardson, and R. L. Urbanke. Finite-length scaling for iteratively decoded ldpc ensembles. *IEEE Transactions on Information Theory*, 55(2):473–498, February 2009.

[11] M. Cunche and V. Roca. Optimizing the error recovery capabilities of ldpc-staircase codes featuring a gaussian elimination decoding scheme. *IEEE International Workshop on Signal Processing for Space Communications*, October 2008.