PAPER *Special Section on Latest Advances in Fundamental Theories of Signal Processing*

Layered Low-Density Generator Matrix Codes for Super High Definition Scalable Video Coding System

Yoshihide TONOMURA[†a)], ***Student Member***, **Daisuke SHIRAI**[†], **Takayuki NAKACHI**[†], **Tatsuya FUJII**[†], ***and*** **Hitoshi KIYA**[††], ***Members***

SUMMARY In this paper, we introduce layered low-density generator matrix (Layered-LDGM) codes for super high definition (SHD) scalable video systems. The layered-LDGM codes maintain the correspondence relationship of each layer from the encoder side to the decoder side. This resulting structure supports partial decoding. Furthermore, the proposed layered-LDGM codes create highly efficient forward error correcting (FEC) data by considering the relationship between each scalable component. Therefore, the proposed layered-LDGM codes raise the probability of restoring the important components. Simulations show that the proposed layered-LDGM codes offer better error resiliency than the existing method which creates FEC data for each scalable component independently. The proposed layered-LDGM codes support partial decoding and raise the probability of restoring the base component. These characteristics are very suitable for scalable video coding systems.

key words: *LDPC codes, LDGM codes, layered coding, scalable video coding, JPEG 2000, multicast streaming, real time streaming, super high definition movies*

1. Introduction

With recent advances in computational power and the broadband infrastructure, the Internet-based video streaming service is receiving a lot of attention. Since the raw digital image contents are too large to transmit economically, video standards have been created to compress the data. The international standards MPEG-2 [1] and H.264/AVC [2] are widely used for digital TV broadcasting service, but unlike TV broadcasting service, Internet broadcasting service must be flexible enough to support the many transmission and end-user environments expected.

In response, several scalable video coding schemes have been introduced to support multicast streaming over IP [3]–[6]. These scalable schemes make it possible for each user to receive an arbitrary image quality. For example, Motion JPEG 2000 standard [4] has extensive resolution/quality scalability. These scalabilities are suitable to treat super high definition (SHD) movies (4096 × 2160 pixels) like digital cinema [7]. Furthermore, thanks to broadband networks, high quality contents are being multicasted using the Motion JPEG 2000 [8]–[10] to the movie theaters and this new form is called "ODS: Other Digital Stuff/ Online Digital Source." ODS is starting to garner attention for new in-theater entertainment including real-time sports, music, pre-show ads and more [11].

In the case of IP-based broadband video streaming, packet losses are frequent. Scalability functions lead to error resiliency because the decoder can process a subset of the original streaming data. Even though scalable video coding schemes are error resilient, errors which cannot be restored do happen occasionally in packet erasure channels. To protect packet data, it is common to use forward error correction (FEC) codes [12]–[15]. Reed-Solomon codes [12] based on algebraic structures are commonly used for multicast streaming. However, these codes make it difficult to lengthen the block length because their decoding processes feature polynomial-time decoding. This feature can be a cause of low coding efficiency and excessive computation overhead if we treat large data such as SHD contents. Thus, FEC based on algebraic structures are not suitable for the emerging applications.

FEC codes based on very sparse matrixes are beneficial because they offer linear-time decoding and make it possible increase correction capacity in packet erasure channels [13], [14]. In particular, Low-Density Parity-Check (LDPC) codes [16], [17] are error correcting codes with a very sparse matrix; they were the first codes to allow data transmission rates close to the theoretical maximum, the Shannon Limit. Moreover, Low-Density Generator Matrix (LDGM) codes [14], [17] are the first codes to allow linear-time encoding and decoding. However, the structure of these sparse codes does not support partial decoding. Therefore, if we use these codes for scalable video data, performance is low, or scalability is lost.

In this paper, we propose efficiently structured LDGM codes that yield scalable video coding systems; we call them the layered-LDGM codes. The layered-LDGM codes create FEC data using one sparse matrix that considers the relationship between the scalable components. The result is that the important components are protected by all received FEC data. In other words, the layered-LDGM codes raise the probability of restoring the important components even though they use only one sparse matrix. Furthermore, the layered-LDGM codes allow partial decoding in which each component can start to be decoded before all block data is received. The performance of these codes are evaluated by the density evolution method [19] and the potential of the

Manuscript received July 10, 2008.
Manuscript revised October 17, 2008.

[†] The authors are with NTT Network Innovation Laboratories, NTT Corporation, Yokosuka-shi, 239-0847 Japan.

[††] The author is with the Department of Information and Communication Systems Engineering, Faculty of System Design, Tokyo Metropolitan University, Hino-shi, 191-0065 Japan.

a) E-mail: tonomura.yoshihide@lab.ntt.co.jp

DOI: 10.1587/transfun.E92.A.798

layered LDGM codes is clarified. The layered LDGM codes are also evaluated when applied to a Motion JPEG 2000 system [10].

This paper is organized as follows. In Sect. 2, we describe the LDGM coding process. Section 3 introduces the layered LDGM codes. Section 4 provides simulation details and compares the performance of the proposed LDGM scheme to the existing LDGM scheme. Section 4 concludes this work by listing future directions.

2. Background

This paper studies the use of a scalable video coding scheme for the multicast streaming of high quality movies. The framework is illustrated in Fig. 1. In this framework, SHD images are compressed by a scalable video coding scheme and each layer is packetized. Each packet is sent to an error correction encoder and FEC packets are generated. All packets are delivered to the appropriate users over the Internet. A key point is that the packets are selectively delivered to meet each user's request. For example, HD users require only base layer and its FEC data. SHD users require, in addition, the enhanced layer and its FEC data. In this way, efficient multicast streaming is realized. More details of the basic technologies are described below.

2.1 Overview of Motion JPEG 2000 [4]

Motion JPEG 2000 (MJ2) is the leading digital film standard because it supports images that have large resolution and high bit-depth, as well as advanced scalability such as SNR and resolution. In this subsection, we describe MJ2 and its data structure.

The image part of an MJ2 file consists of JPEG 2000 (JP2) files. The JP2 coding algorithm is shown in Fig. 2. First, input video images are sent to wavelet decomposition and divided into the subband coefficients. Each subband coefficient is then quantized and sent to EBCOT. In the EBCOT processes, quantized coefficients are divided into each code block partition and each code block coefficient is transformed into binary data. This process is based on occurrence rate. The binary data is compressed by an arithmetic coder and the compressed data is quantized. This quantization process is called post quantization. After these processes, the data is packetized, and the JP2 bit stream is generated. We show an example of an MJ2 code stream in Fig. 3. This progression order is called RLCP (resolution, SNR layer, component, position).

2.2 Internet Burst Packet Erasure Channel Model

This subsection describes the Internet burst packet erasure model. Sometimes, packet loss is a problem for multicast video streaming over the Internet. Packet loss can result from not only network trouble, but also human error and system terminal shortfalls.

It is well known that Internet packet loss is bursty, i.e. packet losses are strongly correlated. Some papers examined models to capture the packet loss behavior of end-to-end Internet communications; most found strong support for the Gilbert model [20], [21]. Therefore, we assume that the Internet follows the Gilbert model.

The Gilbert model is illustrated in Fig. 4. The probabilistic Gilbert model simulates frame deletion based on packet loss and burst loss rate. In this model, Good state and Bad state are defined. While in the Bad state, the transmitted packet is lost. On the other hand, the transmitted packet is surely received in the Good state. In Fig. 4, P_{GB} is the probability that a packet is lost when the previous packet is lost. Likewise, P_{BG} is the probability that a packet will be received when the previous packet is lost. Consequently, av-

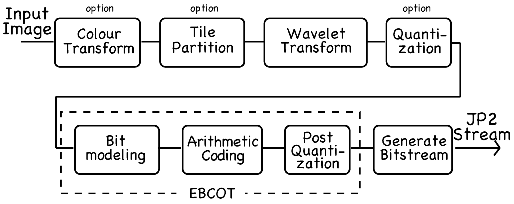


Fig. 2 JPEG 2000 coding algorithm.

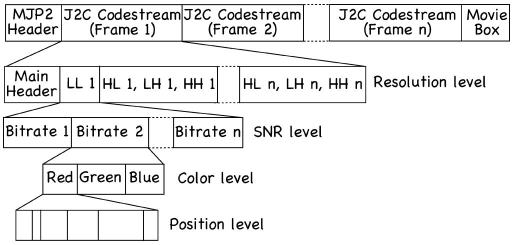


Fig. 3 Example of Motion JPEG 2000 stream. (Progression order RLCP)

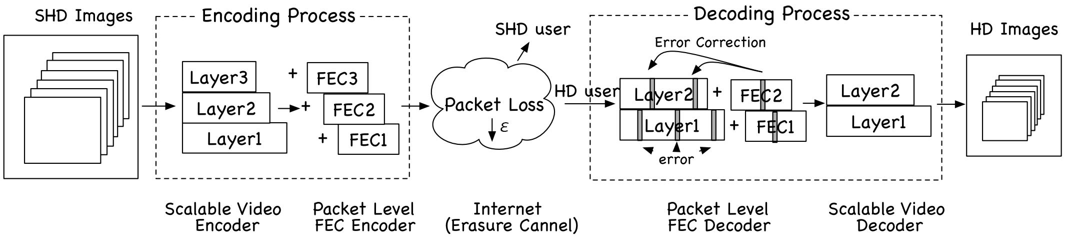


Fig. 1 IP multicast streaming framework for scalable video coding using forward error correction.

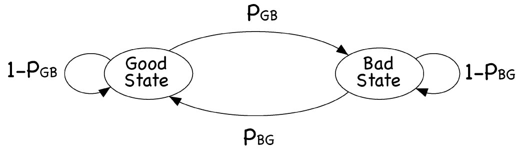


Fig. 4 Gilbert channel model for burst error network.

erage packet loss rate ε and average burst length L are given by

$$\varepsilon = \frac{P_{GB}}{P_{GB}+P_{BG}} \quad \text{and} \quad L = \frac{1}{P_{BG}}. \tag{1}$$

2.3 Low-Density Generator Matrix Codes(LDGM Codes)

To date, many FEC codes have been introduced. Among them, LDGM codes have been shown have high potential and can be advantageously used in many situations [14]. In particular, [14] shows that LDGM codes are more suitable for large data applications than the other FEC codes such as RS codes and LDPC codes. For these reasons, we adopt this code for multicast streaming. In this subsection, we describe LDGM codes.

Since LDGM codes are a variant of the LDPC code, we describe LDPC code to begin with. LDPC codes are a kind of linear code; parity check matrix $\mathbf{H}$ contains mostly 0's and only a small number of 1's. LDPC codes are defined by

$$\mathbf{0} = \left[\mathbf{H}_{m,n}\right]\mathbf{W}_n^t \ (\text{mod } 2) \tag{2}$$

where $\mathbf{H}_{m,n}$ is a large sparse matrix with m columns and n rows and $\mathbf{W}_n$ consists of n received packets. $\mathbf{W}_n$ is a l-by-n matrix, where l is packet length. If the LDPC code is a systematic code, received packet $\mathbf{W}_n$ is given by

$$\mathbf{W}_n = \left[\mathbf{S}_k | \mathbf{C}_m\right] \tag{3}$$

$$= \left[\mathbf{s}_1 \quad \mathbf{s}_2 \quad \ldots \quad \mathbf{s}_k \quad | \quad \mathbf{c}_1 \quad \ldots \quad \mathbf{c}_m\right] \tag{4}$$

where $\mathbf{S}_k$ consists of k original packets and $\mathbf{C}_m$ consists of m parity packets. The sizes of $\mathbf{S}_k$ and $\mathbf{C}_m$ are l-by-k and l-by-m matrices, respectively.

The difference between LDGM and LDPC codes lies in the parity check matrix. For example, the sparse matrix of an LDGM code consists sparse matrix $\mathbf{P}$ and staircase matrix $\mathbf{T}$. The parity check matrix $\mathbf{H}_{m,n}$ is given by

$$\mathbf{H}_{m,n} = \left[\mathbf{P}_{m,k} | \mathbf{T}_{m(staircase)}\right] \tag{5}$$

$$= \left[\begin{array}{ccc|ccccc} & 1 & & 1 & & & & \\ 1 & & & 1 & 1 & & & \\ \cdots & \cdots & \cdots & & \ddots & \ddots & & \\ & & 1 & & & & 1 & 1 \end{array}\right] \tag{6}$$

where $\mathbf{P}_{m,n}$ is a m-by-k sparse matrix made by a random function and $\mathbf{T}$ is a m-by-m square staircase matrix[†].

2.3.1 Encoding Process

Briefly, at the encoder side, the LDGM coder calculates each parity packet as the eXclusive OR (XOR) of sparse matrix $\mathbf{P}$ and the original packet. From Eqs. (2)–(6), the relationship between the FEC packets and the original packets are as follows:

$$\mathbf{0} = \left[\mathbf{H}_{m,n}\right]\mathbf{W}_n^t(\text{mod } 2) \tag{7}$$

$$= \left[\mathbf{P}_{m,k} | \mathbf{T}_m\right]\begin{bmatrix} \mathbf{S}_k^t \\ - \\ \mathbf{C}_m^t \end{bmatrix}(\text{mod } 2) \tag{8}$$

$$= \left[\mathbf{P}_{\mathbf{m},\mathbf{k}}\right]\mathbf{S}_k^t + \left[\mathbf{T}_\mathbf{m}\right]\mathbf{C}_m^t(\text{mod } 2) \tag{9}$$

The FEC packets $\mathbf{C}_m$ can be written as

$$\mathbf{C}_m^t = \left[\mathbf{T}_m^{-1}\right]\left[\mathbf{P}_{m,k}\right]\mathbf{S}_k^t \ (\text{mod } 2) \tag{10}$$

From Eq. (10), the encoding operation consists of two steps: first compute an intermediate parity vector $\mathbf{V}_m = \left[\mathbf{P}_m\right]\mathbf{S}_k^t$; then pass $\mathbf{V}_m$ through an accumulator to create $\mathbf{C}_m$. These processes are written as

$$\begin{array}{ccll} \mathbf{c}_1 & = & & \sum_{j=1}^{k} P_{1,j}\mathbf{s}_j \ (\text{mod } 2) \\ \mathbf{c}_2 & = & \mathbf{c}_1 + & \sum_{j=1}^{k} P_{2,j}\mathbf{s}_j \ (\text{mod } 2) \\ \vdots & = & \vdots & \\ \mathbf{c}_m & = & \mathbf{c}_{m-1} + & \sum_{j=1}^{k} P_{m,j}\mathbf{s}_j \ (\text{mod } 2) \end{array}, \tag{11}$$

where $P_{m,j}$ are the matrix elements of each column. From Eq. (11), the m parity packets $\mathbf{C}_m$ can be computed in linear time.

2.3.2 Decoding Process

The decoding process simply consists of using the belief propagation (BP) algorithm to solve the system of linear Eq. (2). 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 some 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. 5(a). From Fig. 5(a), the check nodes compute the values of r_i as follows:

$$r_i := \begin{cases} \sum_{j \neq i} q_i \ (\text{mod } 2) & \text{if} \quad \forall j \neq i, q_i \neq error \\ error & \text{otherwise} \end{cases} \tag{12}$$

2) **Vertical step.** The vertical step is illustrated in Fig. 5(b). From Fig. 5(b), the variable nodes update the values of q_i as follows:

[†] Some papers describe this code as an irregular repeat-accumulate (IRA) code [18]. In this paper, we define LDGM codes as including a lower triangle matrix($\mathbf{T}$). Therefore, we call this code an LDGM code.

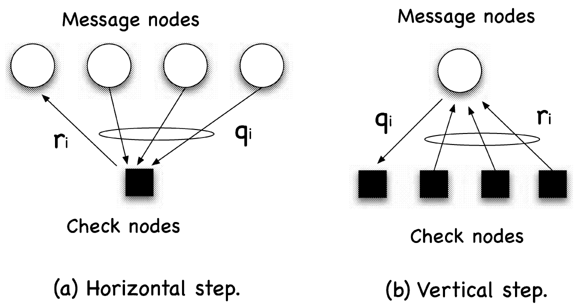


Fig. 5 Message passing rule in horizontal and vertical steps.

$$q_i := \begin{cases} r_j & \text{if} \quad \exists r_j, 1 \le j \le d_v, r_j \ne error \\ error & \quad \text{otherwise} \end{cases} \tag{13}$$

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

3. Layered LDGM Codes

The LDGM codes have the advantage of being able to encode/decode in linear-time. However, the structure of the existing LDGM codes does not support partial decoding. Therefore, if we use these codes for scalable video data, performance is low, or scalability is lost. In this section, we introduce efficient structures for LDGM codes that can realize scalable video coding systems.

3.1 Semi-Random Structure of Layered LDGM Codes

We consider the creation of FEC packets for each layer of data. Simply, each layer uses a different sparse matrix as follows:

$$\begin{aligned} \mathbf{C}^t_{m1} &= \left[\mathbf{T}^{-1}_{m1}\right]\left[\mathbf{P}_{m1,k1}\right]\mathbf{S}^t_{k1} \pmod 2 \\ \mathbf{C}^t_{m2} &= \left[\mathbf{T}^{-1}_{m2}\right]\left[\mathbf{P}_{m2,k2}\right]\mathbf{S}^t_{k2} \pmod 2 \\ \vdots \;&= \qquad\qquad \vdots \\ \mathbf{C}^t_{mL} &= \left[\mathbf{T}^{-1}_{mL}\right]\left[\mathbf{P}_{mL,kL}\right]\mathbf{S}^t_{kL} \pmod 2 \end{aligned} \tag{14}$$

where $\mathbf{C}_{m1} \ldots \mathbf{C}_{mL}$ are FEC packets for base layer $\mathbf{S}_{k1}$ (base layer) and layer 2 … L $\mathbf{S}_{k2} \ldots \mathbf{S}_{kL}$ (expanded layer), and $m1$ is the number of FEC packets for base layer with length $k1$, $m2$ is the number of FEC packets for expanded layer with length $k2$. We call Eq. (14) version the "existing method."

In this case, error resiliency is that determined by adding FEC packets independently. However, it is not so useful to restore just the expanded layers. This is because the expanded layers can not work without the base layer. In the existing method, FEC packets for the expanded layers are effective only for those layers.

Our solution is an efficient structure for more effective LDGM codes (Layered-LDGM Codes). It allows the FEC packets for the expanded layers to be used in regenerating the base (or lower) layer; this increases the total block length and raises the probability of restoring the base layer. The layered-LDGM codes are made by combining several sparse matrixes. The layered-LDGM codes are defined by

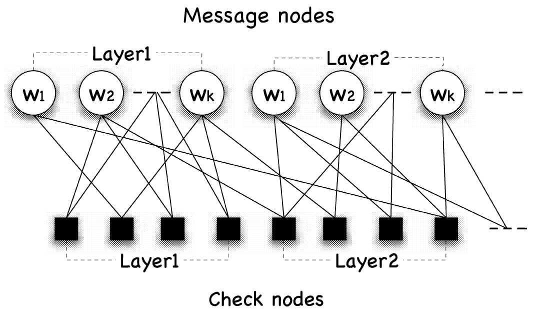


Fig. 6 Corresponding tanner graph of layered-LDGM codes.

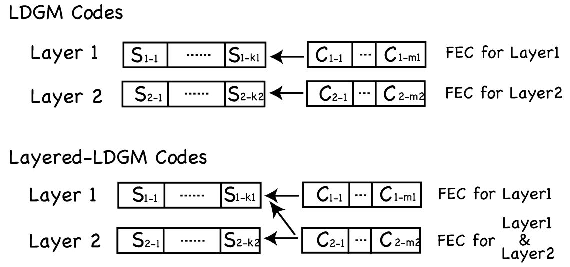


Fig. 7 Relationship between each layer and FEC data.

$$\mathbf{0} = \left[\mathbf{H}_{L\ m,n}\right]\mathbf{W}^t_{L\ n} \tag{15}$$

where the layered sparse matrix which uses layered-LDGM codes and the received packets which include each layer's data $\mathbf{W}_{L\ n}$ are given by

$$\mathbf{H}_{L\ m,n} = \left[\begin{array}{c|c} \begin{array}{c|c} \mathbf{P}_{m1,k1} & \mathbf{0} \\ \hline \mathbf{P}_{m2,k1+k2} & \mathbf{0} \\ \hline \vdots & \mathbf{0} \\ \hline \mathbf{P}_{mL,k1+k2+\cdots+kL} & \end{array} & \mathbf{T}_m \end{array} \right] \tag{16}$$

$$\mathbf{W}_{L\ n} = \left[\mathbf{S}_{L\ k}|\mathbf{C}_{L\ m}\right] = \left[\mathbf{S}_{k1}, \ldots, \mathbf{S}_{kL}|\mathbf{C}_{m1}, \ldots, \mathbf{C}_{mL}\right] \tag{17}$$

The tanner graph of layered-LDGM codes is illustrated in Fig. 6. From Fig. 6, we can see that the received packets for the base layer are connected to those of the expanded layers. The relationship between the layers and FEC packets is illustrated in Fig. 7. From Fig. 7, it is clear that the proposed structure yields the valuable characteristic that the base layer is protected by more parity packets, and so is very suitable for layered video data. Furthermore, the layered-LDGM code can support partial decoding.

The encoding and decoding process of layered-LDGM codes are same as the basic LDGM codes given in Sects. 2.3.1 and 2.3.2. If we use partial decoding, two layer decoding for example, the sparse matrix is given by

$$\mathbf{H}_{L\ m1+m2,n2} = \left[\begin{array}{c|c} \begin{array}{c|c} \mathbf{P}_{m1,k1} & \mathbf{0} \\ \hline \mathbf{P}_{m2,k1+k2} & \end{array} & \mathbf{T}_{m1+m2} \end{array} \right]. \tag{18}$$

Another example, changing the progression order in JPEG 2000 is described in Appendix B.

Note that the proposed structure requires the inclusion of a lower triangle matrix like a staircase matrix because the lower triangle matrix maintains the relationship of each layer from the encoder side to the decoder side. Our method is the first code to focus on this feature and was designed to achieve this goal and fast encoding.

3.2 Array Structure of Layered LDGM Codes

Array LDPC codes were proposed for correcting burst errors and offer easy implementation [22]. In this section, we describe layered-LDGM codes based on an array structure (Layered-ALDGM Codes).

We construct the layered-ALDGM codes by replacing sparse matrixes **P** with circulant permutation matrixes α. The sparse matrix **P** is given by the modified reconstruction rule

$$\mathbf{P_{m,k}} = \begin{bmatrix} I & I & I & \dots & I \\ I & \alpha & \alpha^2 & \dots & \alpha^{(c-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ I & \alpha^{(r-1)} & \alpha^{2(r-1)} & \dots & \alpha^{(c-1)(r-1)} \end{bmatrix} \tag{19}$$

where I is the $q \times q$(q is prime number) identity matrix, c and r are two integers such that $c, r \leq q$ and α is the permutation matrix defined by

$$\alpha = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 0 & 0 & \dots & 0 \end{bmatrix}. \tag{20}$$

The layered-ALDGM codes are made by substituting Eq. (19) and Eq. (20) into Eq. (18). As a result, the layered-ALDGM codes does not have 4-cycle same as array LDPC codes.

4. Simulation Results

We evaluated the performance of our method by using the density evolution (DE) method [19] and then implemented it in a JPEG 2000 system.

4.1 Evaluation of Threshold Using DE Method

The threshold shows the asymptotic performance of any LDPC code. In this subsection, we evaluate the thresholds of layered-LDGM codes using the DE method with several specific degree distributions. The DE method is described in Appendix A.

For brevity, we evaluate regular layered-LDGM codes with rate 10[%]. In general, regular LDPC codes have a regular parity variable matrix in which all variable nodes and check nods have the same degree. Regular layered-LDGM codes, on the other hand, the sparse matrixes in which all variable nodes have the same degree. For example, a regular layered-LDGM code for two layers has two sparse matrixes **P** in which all variable nodes have degree 3. This means, the regular layered-LDGM codes consist of all sparse matrixes in which all variable nodes have the same degree.

If a regular layered-LDGM code (variable nodes 3) treats data with the same layer length, which means that the base layer and expanded layer have the same length, the regular layered-LDGM code with rate 10[%] is written as

$$\lambda(x) = 0.0471x + 0.3176x^2 + 0.6353x^5 \tag{21}$$

$$\rho(x) = 0.3412x^{28} + 0.6588x^{55} \tag{22}$$

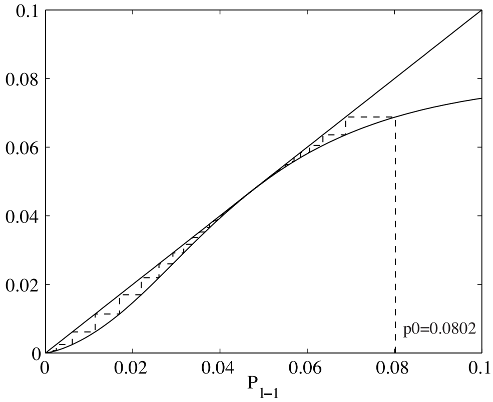


Fig. 8 Error probability versus decoding iteration number.

Table 1 Relationship between threshold and degree distribution pairs.

	Regular LDPC	k_1:k_2=1:1	k_1:k_2=2:1	k_1:k_2=1:2	k_1:k_2:k_3=1:1:1
Threshold	0.0828	0.0802	0.0784	0.0835	0.0739
λ_2		0.0471	0.0426	0.0526	0.0357
λ_3	1.0000	0.3176	0.1915	0.4737	0.1607
λ_6		0.6353	0.7660	0.4737	0.3214
λ_9					0.4821
ρ_{29}		0.3412	0.4113	0.2544	0.1726
ρ_{30}	1.0000				
ρ_{42}				0.3684	
ρ_{43}				0.3772	
ρ_{56}		0.6588			0.3333
ρ_{82}			0.5887		
ρ_{83}					0.4940

where λ and ρ are the distribution pair of the variable node degree and the check node degree, respectively. They are define by Eq. (A· 1).

From Eqs. (21), (22) and (A· 4), we can calculate the error probability after the l-th iteration. In Fig. 8, we illustrate the relationship between error probability and decoding iteration number. In this case, the probability converged to 0 after the l-th iteration, which means that this layered-LDGM code has the potential to correct up to $p_0 = 0.0802$. Similarly, we calculated the thresholds of layered-LDGM code with several specified degree distributions; the results are shown in Table 1. It shows that the proposed layered-LDGM codes have good potential to correct error, approaching the performance of regular LDPC codes, while offering partial decoding.

4.2 Evaluation in a JPEG 2000 System

4.2.1 Experimental Setup

We considered a real-time high-quality video distribution service that transfers Super High Definition (SHD) images [10]. Two image sizes were assumed: SHD class called 4 K (4096 × 2160 pixels) and High Definition class called full HD (2048 × 1080 pixels). The original material was compressed by JPEG 2000 and processed to yield a two layer streaming service. Layer1 supports HD users while the expanded layer is for SHD users. The base layer bitrate was set to 85 [Mbps] and the expanded layer bitrate was set to 170 [Mbps]). In other words, HD users receive about 85 [Mbps] video data and SHD users receive about 255 [Mbps] video data. This bitrate almost matches the digital cinema specification.

If each packet is 1500 [bytes], code block lengths $k1, k2$ are set to 450 and 900, respectively. FEC overhead is 10[%] for each layer using layered LDGM codes; $m1$ and $m2$ are set to 45 and 90, respectively. However, it is clear that the error resiliency is high if we use large block lengths and it is important that we use LDGM codes. Therefore, we bind 3 frames together. Consequently, we set the LDGM parameters as follows: $k1 = 1350, k2 = 2700, m1 = 135, m2 = 270$.

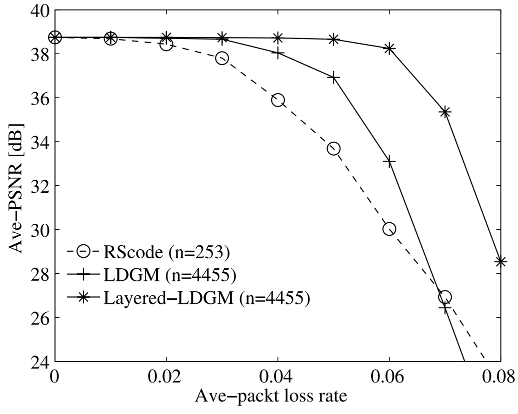


Fig. 9 Characteristics of PSNR and packet loss rate for 4k users. The LDGM and the layered-LDGM consisted of a semi-random structure. ($L = 3$)

4.2.2 Experimental Results

We tested the error resiliency performance of our proposal

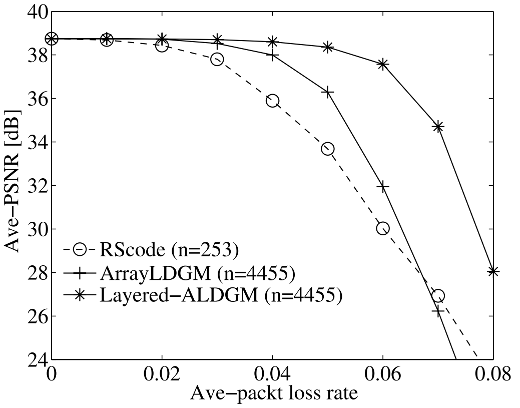


Fig. 10 Characteristics of PSNR and packet loss rate for 4k users. The LDGM and the layered-LDGM consisted of a array structure. ($L = 3$)

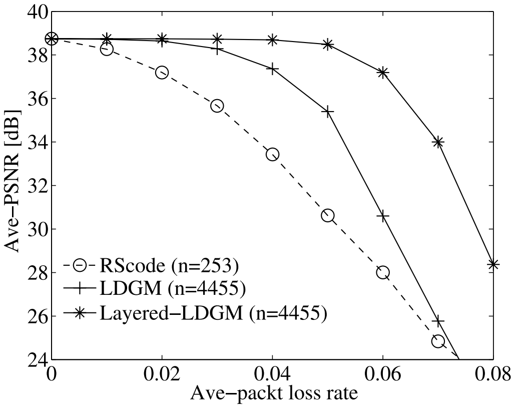


Fig. 11 Characteristics of PSNR and packet loss rate for 4k users. The LDGM and the layered-LDGM consisted of a semi-random structure. ($L = 5$)

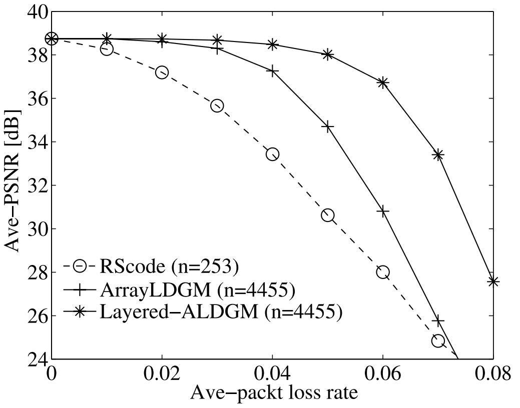


Fig. 12 Characteristics of PSNR and packet loss rate for 4k users. The LDGM and the layered-LDGM consisted of a array structure. ($L = 5$)

using StEM [7], public content created for digital cinema evaluation. Figures 9–12 plot the characteristics of PSNR and packet loss rate for 4k SHD users. Figures 9 and 11 are for semi-random layered-LDGM codes and Figs. 10 and 12 are for layered-ALDGM codes. The vertical axis plots average PSNR of 2000 frames; the packet loss rate (horizontal axis) was generated based on the Gilbert model for the Internet. We set burst error length to $L = 3$ in Figs. 9–10, and $L = 5$ in Figs. 11–12, respectively.

It can be seen that the proposed method offers better error resiliency than the existing methods regardless of the rate. In particular, it should be noted that the proposed layered-LDGM codes show better performance than the simple unlayered LDGM code. We investigate the reason for the excellent error resiliency of our method. Figure 13 plots the decoding success probability for the base layer vs average packet loss rate. The probability of 1 means that base layer decoding never fails. From Fig. 13, the proposed method protected the important base layer more than the existing methods. In other words, the proposed method increases error resiliency of the important parts.

To show an example of the the estimated PSNR distribution, we plot the PSNR of each frame in Fig. 14 at a packet loss rate 4[%], semi-random structure, $L = 5$. As can be seen, the proposed layered-LDGM corrects all packet loss, unlike RS codes and existing LDGM codes which some-

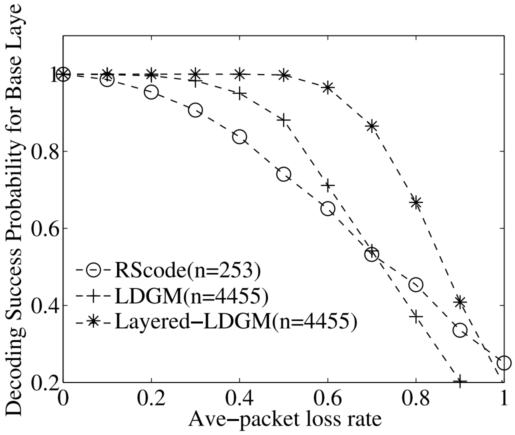


Fig. 13 Decoding success probability for base layer vs average packet loss rate. (Semi-random structure, $L = 5$)

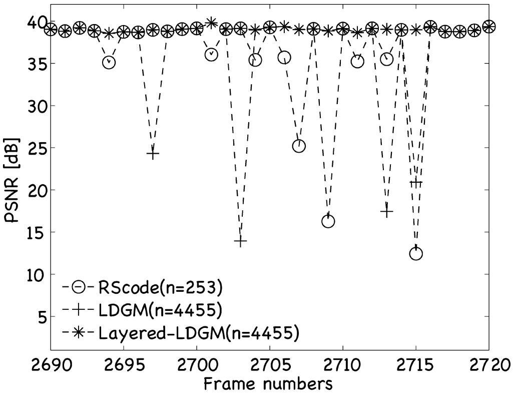
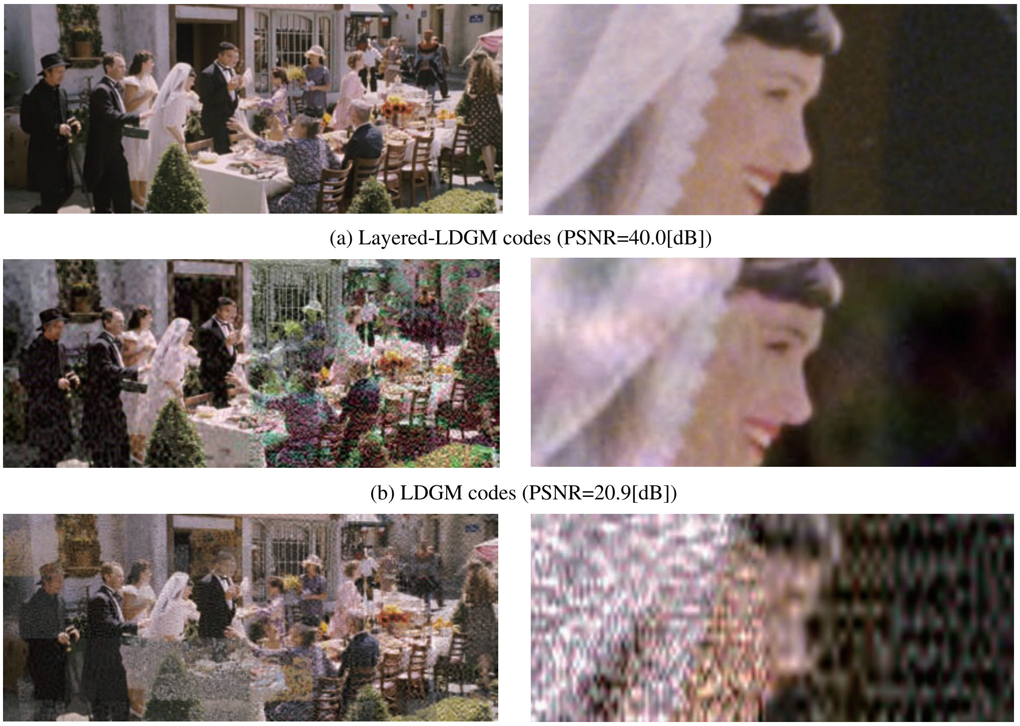


Fig. 14 Frame by frame PSNR for sequence “StEM.” ($L = 5$)



(a) Layered-LDGM codes (PSNR=40.0[dB])

(b) LDGM codes (PSNR=20.9[dB])

(c) RS codes (PSNR=12.4[dB])

Fig. 15 Reconstructed image of StEM 2715 in Fig. 14.

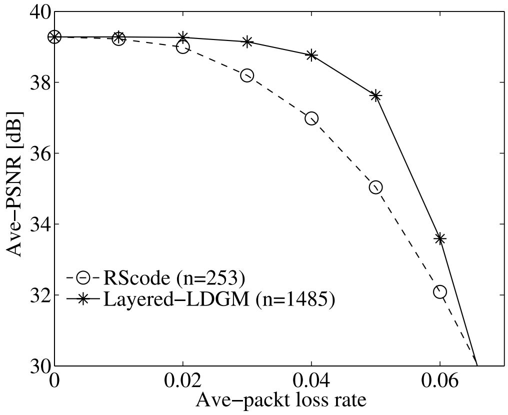


Fig. 16 Characteristics of PSNR and packet loss rate for HD users. The LDGM and the layered-LDGM consisted of a semi-random structure. ($L = 3$)

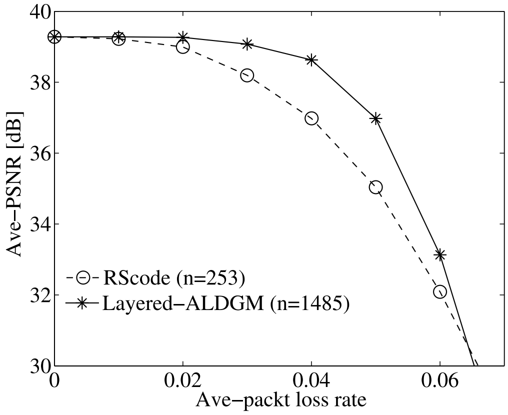


Fig. 17 Characteristics of PSNR and packet loss rate for HD users. The LDGM and the layered-LDGM consisted of a array structure. ($L = 3$)

times fail to correct an error. These errors result in highly visible flicker artifacts. We also show the reconstructed images in Fig. 15, at frame 2715 in Fig. 14. For this frame, unlike the LDGM codes and RS codes which failed to correct both layers, the layered LDGM codes corrected all packet losses. Figure 15 confirms that it is important to protect the base layer.

A comparison of Figs. 9 and 10, shows that the difference in characteristics is small. The difference is due to Internet burst error length. Figures 11 and 12 show same result. From these figures, it can be concluded that our layered-LDGM codes are not sensitive to burst error length. Figures 11 and 12 evaluate the performance of layered-ALDGM codes. From these figures, it can be seen that the layered-ALDGM codes are as effective as the semi-random layered-LDGM codes. The array structure has many benefits such as small circuit size, and the benefits become more valuable as the block length increases.

Figures 16 and 17 plot the characteristics of PSNR and packet loss rate for HD users using the semi-random layered-LDGM codes and layered-ALDGM codes, respectively. It can be seen that both proposed methods offer better error resiliency than the existing RS code, regardless of the rate.

The layered-LDGM codes basically support any layered coding data form such as resolution scalability, quality scalability, and temporal scalability. However, we acknowledge that the proposed method may need to be optimized for each specification. For example, if block length falls below 1000, the proposed method does not offer superior error resiliency. The goal of creating a position optimal design method for layered-LDGM codes is a future task.

5. Conclusion

We presented novel layered low-density generator matrix (LDGM) codes for scalable video coding. The proposed layered LDGM codes are constructed by just one sparse matrix and do not sacrifice scalability. Moreover, the layered LDGM codes enable lost packet data to be recovered effectively before all data is received, which enables progressive decoding. Two variants for layered LDGM codes were introduced: semi-random and array structured. Simulations showed that the proposed method offers better error resiliency than the existing method which creates FEC data for each layer independently. Our proposed structure supports partial decoding and raises the probability of restoring the base layer. Its characteristics are very suitable for scalable video coding. Future work includes developing an optimal design method for these layered LDGM codes.

References

[1] ITU-T Rec. H.262, ISO/IEC 13818-2, “Generic coding of moving pictures and associated audio,” International Standard, March 1995.
[2] Advanced Video Coding for Generic Audiovisual Services, ITU-T Rec. H.264, 2003.
[3] ISO/IEC 15444-1, “Information technology — JPEG 2000 image coding system–Part1: Core coding system,” International Standard, Jan. 2001.
[4] ISO/IEC 15444-3, “Information technology — JPEG 2000 image coding system–part3: Motion JPEG 2000,” International Standard, July 2002.
[5] ISO/IEC 14496-2, “Information technology — Coding of audio-visual objects–Part 2,” International Standard, 2003.
[6] T. Wiegand, G. Sullivan, J. Reichel, H. Schwarz, and M. Wien, “Joint draft 10 of SVC amendment,” Document of ISO/IEC JTC1/SC29/ WG11 and ITU-T SG16 Q.6, JVT-W201, April 2007.
[7] Digital Cinema Initiatives(DCI), http://www.dcimovies.com
[8] S. Fossel, G. Fottinger, and J. Mohr, “Motion JPEG2000 for high quality video systems,” IEEE Trans. Consum. Electron., vol.49, no.4, pp.787–791, Nov. 2003.
[9] T. Shimizu, D. Shirai, H. Takahashi, T. Murooka, K. Obana, Y. Tonomura, T. Inoue, T. Yamaguchi, T. Fujii, N. Ohta, S. Ono, T. Aoyama, L. Herr, N. Osdol, X. Wang, M.D. Brown, T.A. DeFanti, R. Feld, J. Balser, S. Morris, T. Henthorn, G. Dawe, P. Otto, and L. Smarr, “International real-time streaming of 4K digital cinema,” Future Gener. Comput. Syst. vol.22, no.8, pp.929–939, 2006.
[10] D. Shirai, T. Yamaguchi, T. Shimizu, T. Murooka, and T. Fujii, “4K SHD real-time video streaming system with JPEG 2000 parallel codec,” IEEE Asia Pacific Conference on Circuits and Systems APCCAS, pp.1855–1858, Dec. 2006.
[11] S. Ono, “Further use of digital cinema systems (ODS),” The Institute of Image Information and Television Engineers, vol.61, no.5,

pp.591–595, 2007.
[12] L. Rizzo, "Effective erasure codes for reliable computer communication protocols," Comput. Commun. Rev., vol.27, no.2, pp.24–36, April 1997.
[13] M. Luby, "LT codes," Foundations of Computer Science, 2002. Proc. 43rd Annual IEEE Symposium on 16-19, pp.271–280, Nov. 2002.
[14] V. Roca and C. Neumann, "Design, evaluation and comparison of four large block FEC codecs, LDPC, LDGM, LDGM staircase and LDGM triangle, plus a Reed-Solomon small block FEC codec," INRIA, 2004.
[15] A. Shokrollahi, "Raptor codes," IEEE Trans. Inf. Theory, vol.52, no.6, pp.2551–2567, June 2006.
[16] R.G. Gallager, "Low density parity check codes," in Research Monograph series, MIT Press, Cambridge, 1963.
[17] D.J.C. MacKay, "Good error-correcting codes based on very sparse matrices," IEEE Trans. Inf. Theory, vol.45, no.2, pp.399–431, March 1999.
[18] H. Jin, A. Khandekar, and R. McEliece, "Irregular repeat accumulate code," 2nd Int. Symp. on Turbo Codes and Related Topics, pp.1–8, Sept. 2000.
[19] T. Richardson, M.A. Shokrollahi, and R.L. Urbanke, "Design of capacity-approaching irregular low-density parity-check codes," IEEE Trans. Inf. Theory, vol.47, no.2, pp.619–637, 2001.
[20] M. Arai, A. Chiba, and K. Iwasaki, "Measurement and modeling of burst packet losses in internet end-to-end communications," Proc. IEEE Pacific Rim International Symposium on Dependable Computing, pp.260–267, 1999.
[21] W. Jiang and H. Schulzrinne, "Modeling of packet loss and delay and their effect on real-time multimedia service quality," NOSSDAV 2000, June 2000.
[22] J.L. Fan, "Array codes as low-desity parity-check codes," Proc. 2nd Int. Symp. Turbo Codes, Brest, pp.543–546, France, Sept. 2000.

Appendix A: Density Evolution [19]

Density evolution is a tool to evaluate the performance of LDPC codes. The density evolution method analyzes "error" probability in each degree. If the graph has degree distribution pair (λ, ρ), the variable node degree distribution and the check node degree distribution are given by

$$\lambda(x) := \sum_{i}^{dv} \lambda_i x^{i-1} \text{ and } \rho(x) := \sum_{i}^{dc} \rho_i x^{i-1} \qquad \text{(A· 1)}$$

where d_v and d_c are the maximum variable degree and check degree, respectively.

After the lth iteration, the error probability $P(r_i = error)$ in Eq. (12) and the error probability $P(q_i = error)$ in Eq. (13) are given by

$$p_{hl} = 1 - \sum_{k} \rho_k (1 - pv_{l-1})^{k-1} \qquad \text{(A· 2)}$$

$$p_{vl} = p_0 \sum_{d} \lambda_d ph_{l-1}^{d-1} \qquad \text{(A· 3)}$$

where p_0 is the packet loss rate.

From Eqs. (A· 2)–(A· 3), the fraction of erased messages which are passed in the lth iteration, call it p_l, evolves as

$$p_l = p_0 \lambda(1 - \rho(1 - p_{l-1})), \qquad l \geq 0 \qquad \text{(A· 4)}$$

We use the density evolution method to evaluate the proposed method in Sect. 4.1.

Appendix B: Modification for Progression Order Change

The layered-LDGM codes basically support a change in progression order. However, it requires slight modification of the parity check matrix. For example, if we change RLCP progression to CPRL progression, we need to split each matrix. The parity check matrix that allows low resolution image decoding for RLCP progression is given by Eq. (18). On the other hand, the parity check matrix which allows low resolution image decoding for CPRL progression is given by

$$\mathbf{H}_{L\ m,n} = \left[\begin{array}{c|c}
\begin{array}{ccc}
\mathbf{P}_{m1,k1'} & \multicolumn{2}{|c}{\mathbf{0}} \\ \hline
\multicolumn{3}{c}{\mathbf{P}_{m2',k1+k2}} \\ \hline
\mathbf{0} & \multicolumn{1}{|c|}{\mathbf{P}_{m1,k1''}} & \mathbf{0} \\ \hline
\multicolumn{3}{c}{\mathbf{P}_{m2'',k1+k2}} \\ \hline
\multicolumn{2}{c|}{\mathbf{0}} & \mathbf{P}_{m1,k1'''} \\ \hline
\multicolumn{3}{c}{\mathbf{P}_{m2''',k1+k2}}
\end{array} & \mathbf{T}'_m
\end{array}\right] \qquad \text{(A· 5)}$$

$$\mathbf{T}'_m = \begin{bmatrix}
\mathbf{T}_{m1} & & & & \mathbf{0} & \\
 & \mathbf{T}_{m2'} & & & & \\
 & & \mathbf{T}_{m1} & & & \\
 & & & \mathbf{T}_{m2''} & & \\
 & \mathbf{0} & & & \mathbf{T}_{m1} & \\
 & & & & & \mathbf{T}_{m2'''}
\end{bmatrix} \qquad \text{(A· 6)}$$

The relationship between the sparse matrix **P** in Eq. (18) and the sparse matrix **P** in Eq. (A· 5) are written as

$$\mathbf{P}_{m1,k1} = \left[\ \mathbf{P}_{m1,k1'}\ \middle|\ \mathbf{P}_{m1,k1''}\ \middle|\ \mathbf{P}_{m1,k1'''}\ \right] \qquad \text{(A· 7)}$$

$$\mathbf{P}_{m2,k1+k2} = \left[\begin{array}{c} \mathbf{P}_{m2',k1+k2} \\ \hline \mathbf{P}_{m2'',k1+k2} \\ \hline \mathbf{P}_{m2''',k1+k2} \end{array}\right] \qquad \text{(A· 8)}$$

The parity check matrix in Eq. (A· 5) allows partial decoding the same as Eq. (18). Since the effect of this modification is very small, the error resiliency performance is almost the same as that with RLCP progression. In other words, even if we use CPRL progression, we are able to decode only low frequency components in the same way as with RLCP progression. In this way, the proposed layered-LDGM code supports a change in progression.

Yoshihide Tonomura received his B.S. and M.S. degrees in electronics engineering from Nagaoka University of Technology in 2002 and 2004, respectively. He joined NTT Network Innovation Laboratories in 2004. His research interest is focused on image processing theories and applications.

Daisuke Shirai received his B.Eng. degree in electrical engineering and M.Eng. degree in computer science from Keio University, Tokyo, Japan, in 1999 and 2001, respectively. He joined NTT Network Innovation Laboratories, Yokosuka, Japan in 2001 and has been researching super high definition digital cinema systems and their transmission technology. His interests include terabit-level image transmission and scalable video distribution.

Takayuki Nakachi received a Ph.D. degree in electrical engineering from Keio University in 1997. Since he joined NTT Laboratories in 1997, He has been engaged in research on Super High Definition (SHD) image coding, especially in the area of lossless and near-lossless coding. For recent years, he has been researching scalable image/video coding in order to distribute the SHD image contents. From 2006 to 2007, he had been a visiting scientist at Stanford University, California. He is currently a senior research engineer of media processing systems research group in NTT Network Innovation Laboratories. He is a member of IEEE.

Tatsuya Fujii received his B.S., M.S. and Ph.D. degrees, all in electrical engineering from the University of Tokyo, Tokyo, Japan, in 1986, 1988, and 1991, respectively. He joined NTT, Japan, in 1991. He has been researching parallel image processing and super-high-definition image communication networks. In 1996, he was a visiting researcher of Washington University in St. Louis. He is currently a group leader of media processing systems research group in NTT Network Innovation Laboratories. He is a member of ITE of Japan and IEEE.

Hitoshi Kiya received his B.E. and M.E. degrees from Nagaoka University of Technology, Japan and a Dr.Eng. degree from Tokyo Metropolitan University, Tokyo, in 1980, 1982 and 1987 respectively, all in electrical engineering. In 1982, he joined Tokyo Metropolitan University where he is currently a Professor at the Graduate School of System Design. From 1995 to 1996 he attended the University of Sydney, NSW, Australia as a Visiting Fellow. His research interests are in the areas of multirate systems and image processing, including filter bank design and theory, image and video coding, image watermarking and data hiding, and subband adaptive filtering. He received an Excellent Paper Award in 2008 from IEICE. He served as an associate editor of IEICE Transactions Fundamentals and IEEE Transactions on Signal Processing from 1998 to 2002 and from 1998 to 2000, respectively. He was also the Guest Editor-in-Chief for the Special Issues of IEICE Transactions published in 1999, 2006 and 2007 respectively. He was the Technical Committee Chair of Digital Signal Processing of IEICE from 2006 to 2007 and the General Chair of the Japan DSP Educator Conference from 2005 to 2007. He currently serves as a Vice President of IEICE Engineering Science Society and the Editor-in-Chief of IEICE ESS Magazine.