Multicast Communications with Reed Solomon/ Network Joint Coding in Wireless Multihop Networks

Takahiro Matsuda and Tetsuya Takine
Graduate School of Engineering, Osaka University, Osaka, JAPAN
Email: {matsuda, takine}@comm.eng.osaka-u.ac.jp

***Abstract*—In wireless multihop networks, multihop packet transmissions over error-prone wireless links cause significant performance degradation. In this paper, we study a multicast system in wireless multihop networks with Forward Error Correction (FEC) for packet erasures. Although FEC is a powerful tool to recover packet erasures, it has an inherent problem that burdens the network with overhead due to redundant packets. In order to solve the problem, we propose a new multicast system with** *Reed Solomon/network joint coding*. **In the proposed system, information packets from a source node are encoded by** *Reed Solomon erasure* **(RSE) coding and transmitted into the network. At intermediate nodes on multicast paths, packets arriving from different links are encoded by** *linear network coding* **(LNC). The joint coding provides highly robust and efficient multicast communication because RSE coding provides a recovery mechanism from packet erasures and LNC reduces the number of relayed packets in the network. From the fact that both RSE coding and LNC are linear coding, we propose a new decoder for the joint coding. In the proposed decoder, a** *decoding matrix* **is constructed by combining the parity matrix of RSE coding and a coding matrix of LNC. Destination nodes retrieve information packets by solving a system of simultaneous equations constructed by the decoding matrix. Simulation experiments show that the joint coding provides highly robust and efficient multicast communications.**

***Index Terms*—linear network coding, erasure correction code, Reed Solomon code, joint coding, multicast, wireless multihop networks**

I. Introduction

Recently, there have been a lot of research efforts in wireless multihop networks including mobile ad hoc networks (MANETs), wireless sensor networks (WSNs), and wireless mesh networks. In wireless multihop networks, error control techniques such as automatic repeat request (ARQ) and forward error correction (FEC) are very important because multihop packet transmissions over error-prone wireless links cause significant performance degradation.

In this paper, we study FEC in order to improve the performance degradation. In general, FEC has two roles, *error correction* and *erasure correction*. Packet errors mean receiving packets with incorrect data, and they are handled at physical and data link layers. On the other hand, packet erasures mean missing packets, and they are handled at network, transport, and application layers. In this paper, we focus on erasure correction implemented in application layer and apply FEC to a multicast system in wireless multihop networks.

FEC provides significant performance improvements of multicast in both realtime streaming services and reliable file transfer services. In the streaming services, it enhances the quality of streaming media without delay. In the reliable file transfer services, it resolves the feedback implosion problem due to feedback information for retransmission of packets, i.e., acknowledgements (ACKs) or negative acknowledgements (NAKs). However, FEC has an inherent problem that burdens the network with overhead due to redundant packets (i.e., parity packets).

In this paper, we propose a new multicast system with *Reed Solomon erasure (RSE)* / network joint coding. In the proposed system, source nodes encode information packets with RSE coding [15], [17] and transmit them into the network. At intermediate nodes on multicast paths, packets arriving from different links are encoded by *linear network coding (LNC)* [1], [11]. Because RSE coding provides a recovery mechanism from packet erasures and LNC reduces the number of relayed packets in the network [14], [16], the proposed system is expected to provide highly robust and efficient multicast communications.

RSE coding is based on Reed Solomon burst error correction coding and it is a kind of linear coding. Although the joint coding can be constructed by an arbitrary linear erasure correction coding, we use RSE coding because it is one of typical erasure correction coding and has been utilized in many network applications.

On the other hand, when LNC is used, we must consider two important technical issues: assigning valid coding vectors to intermediate nodes and establishing disjoint paths from source nodes to destination nodes. If invalid coding vectors are assigned, destination nodes cannot decode received packets successfully. In static network environments, where network topology does not change, valid coding vectors can be assigned to all intermediate nodes. For example, in [24], valid coding vectors are assigned to all intermediate nodes by exchanging coding information among neighboring nodes. In dynamic network environments, where network topology changes dynamically, decentralized coding vector assignment of

Manuscript received March 30, 2009; accepted July 14, 2009.

doi:10.4304/jcm.4.11.856-864

coding vectors such as Randomized Network Coding [8] is required. Although the decentralized assignment causes failure of decoding, the probability of decoding failure can be made to be negligibly small by constructing coding vectors on finite field of sufficiently large order. In this paper, we assume a static network environment. For the latter issue, although several research efforts for decentralized or optimal routing algorithms have been studied [12], [21], we use a simple routing algorithm based on a vertex disjoint multiple paths algorithm [2].

In the proposed system, information packets are encoded successively with RSE coding and LNC, which are defined on the same finite field, and delivered to destination nodes. We propose a new decoder for these packets. Generally, in linear block codes such as RSE coding, coded packets lie in the linear sub-space generated by the parity matrix. Therefore, at a destination node, a *decoding matrix* can be constructed by combining the parity matrix and a coding matrix for LNC, because RSE coding and LNC are defined on the same finite field. The destination node retrieves information packets by solving a system of simultaneous equations constructed by the decoding matrix. The proposed decoder simplifies the decoding algorithm, because decoding algorithms for RSE coding and LNC are combined into a single decoding algorithm.

Although the joint coding is equivalent to network erasure correction coding [9], [19], these two coding mechanisms are completely different from the viewpoint of protocol layering. While the network erasure correction coding is implemented in a network layer, the proposed system has two coding mechanisms in two different layers, i.e., RSE coding in application layer and LNC in network layer. Therefore, in order to design the network erasure correction coding, network layer requires cross layer information such as QoS (Quality of Services) information of application layer. On the other hand, in order to design the proposed system, network layer requires cross layer coding information such as coding parameters of RSE coding (code length, order of finite field, etc.). When FEC mechanisms are not implemented in the application layer of multicast systems, the network erasure correction coding would be appropriate. If a FEC mechanism has been already implemented in application layer of multicast systems, however, the approach of the proposed system should be natural because the design of application layer need not be modified.

The rest of this paper is organized as follows. In section II, we briefly review related works. In section III, we explain the system model and describe an overview of the proposed multicast system. In section IV, we explain RSE coding and LNC, and propose a decoder with decoding matrices. In section V, we explain the routing algorithm used in this paper. In section VI, we evaluate the proposed system with simulation experiments. Finally, concluding remarks are provided in section VII.

II. Related Works

We refer to coding mechanisms based network coding for packet errors as network error correction coding, and to those for packet erasures as network erasure correction coding. In this section, we explain them briefly.

Network error correction coding mechanisms are studied in [6], [7], [10], [20], [23]. Packet errors occur due to bit errors in noisy wireless channels and Byzantine adversaries that malicious nodes forward corrupted packets to destinations. In network error correction coding mechanisms, the source generates a set of packets and transmits them on edge-disjoint paths. Destination nodes can correct packet errors as follows. Let $\boldsymbol{x}$ and $\boldsymbol{y}$ denote a set of packets transmitted from a source node and a set of packets received at a destination node, respectively. When LNC is used, $\boldsymbol{y}$ can be represented by $\boldsymbol{y} = \boldsymbol{x}\boldsymbol{A} + \boldsymbol{e}\boldsymbol{F}$, where $\boldsymbol{e}$ denotes errors, and $\boldsymbol{A}$ and $\boldsymbol{F}$ are matrices depending on LNC. $\boldsymbol{x}$ are retrieved by estimating $\boldsymbol{x}^*$ with the minimum Hamming distance from $\boldsymbol{y}$, that is, $\boldsymbol{x}^* = \arg\min_{\boldsymbol{x}} \ (\boldsymbol{y}, \boldsymbol{x}\boldsymbol{A})$, where $(\boldsymbol{a}, \boldsymbol{b})$ represents the Hamming distance between $\boldsymbol{a}$ and $\boldsymbol{b}$ [6]. In a network error correction coding mechanism proposed in [23], before transmitting packets, source nodes encode them with a product code, which is composed of different two error correction codes. Intermediate nodes encode received packets with LNC and forward them. Destination nodes decode the received packets with an iterative decoding algorithm based on the Turbo decoding. Our proposed system uses a similar approach to this mechanism because in the proposed system, source nodes encode packets with RSE coding before transmitting them.

Network erasure correction coding mechanisms are studied in [9], [19]. Packet erasures occur due to buffer overflow at congested links, contention of frames in MAC layer, and link failure caused by node mobility. In [9], the general framework for LNC and a recovery mechanism from link failures are described as follows. Let $\boldsymbol{x} = (x_1 \; x_2 \; \cdots \; x_K)$ denote information packets generated from a source node. At the source node, $\boldsymbol{x}$ is transformed linearly to $\boldsymbol{y} = \boldsymbol{x}\boldsymbol{B}$ with a $K \times$ matrix $\boldsymbol{B}$. Packets $\boldsymbol{y}$ are transmitted to the network and encoded by LNC at intermediate nodes. Packets $\boldsymbol{z}$ received at a destination node can be represented by $\boldsymbol{z} = \boldsymbol{y}\boldsymbol{A}$, where $\boldsymbol{A}$ denotes a transfer matrix. $\boldsymbol{x}$ is retrieved by solving a system of simultaneous equations $\boldsymbol{y} = \boldsymbol{x}\boldsymbol{B}\boldsymbol{A}$. If some packets are lost on their paths, $\boldsymbol{A}$ is rewritten according to the location of the packet erasure. If $\boldsymbol{A}$ has rank of K against any $- K$ packet erasures, $\boldsymbol{x}$ can be always retrieved successfully. While in [9] and this paper, multicast communications are targeted, a recovery mechanism from link failures for unicast communications is studied in [19].

As will be explained in the next section, the joint coding in the proposed system corresponds to network erasure correction coding that $\boldsymbol{B}$ is set to be a generator matrix of RSE coding. Therefore, it belongs to a class of network erasure correction coding. However, as described in section I, the proposed system is based on the idea

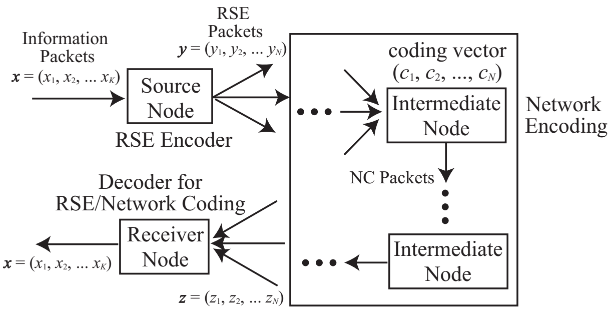

Fig. 1. Multicast system with RSE coding and LNC.

of cross layer coding information, where erasure coding information in application layer and network coding information in network layer are combined. To the best of our knowledge, such a multicast system with network erasure correction coding has not been proposed so far. In [9], [19], the frameworks for network erasure correction are shown. However, erasure correction capability of network erasure correction coding is not evaluated in practical network environments and its advantages over general erasure coding without network coding are not discussed. We consider that network erasure correction coding has an important advantage of providing the erasure correction capability with reduced network resource usage in multicast networks. In order to evaluate this advantage, we introduce *network code rate* in section VI.

III. System Model

In this paper, nodes are classified into three types, source nodes, intermediate nodes, and destination nodes. We study a single-source multicast communication in wireless multihop networks, where all nodes are fixed and one source node transmits data to multiple destination nodes. Each node has an omni-directional antenna and communicates with nodes within communication range R. We consider only packet erasures and assume that no packet error happens in the network.

Fig. 1 shows the proposed system. We define *information packets* as original packets that the source node tries to deliver to its destination nodes, *RSE packets* as RSE encoded information packets at the source node, and *NC packets* as network encoded RSE packets at intermediate nodes.

According to requirements for erasure correction capability, the source node first selects coding parameter $(\ ,K)$, which means that $(\ -K)$ redundant packets for erasure correction are added to every K information packets. The proposed system can retrieve all information packets even when $(\ -K)$ or less packets are lost. At the source node, all information packets are separated into sets of K information packets. Next, disjoint paths from the source node to each destination node are established. In section V, we will explain a vertex disjoint multiple paths algorithm for network coding used in this paper. The routing algorithm is based on a link-state routing algorithm and each node broadcasts its link state information with a flooding algorithm such as a multipoint relay scheme [4].

Let $\boldsymbol{x} = (x_1\ x_2\ \cdots\ x_K)$ denote a set of information packets. The source node encodes $\boldsymbol{x}$ into RSE packets $\boldsymbol{y} = (y_1\ y_2\ \cdots\ y_N)$ $(\ \geq K)$ with RSE encoder. RSE packets are transmitted on the established paths and intermediate nodes with several input links on the paths encode the RSE packets into NC packets by combining received packets linearly. Destination nodes receives NC packets $\boldsymbol{z} = (z_1\ z_2\ \cdots\ z_N)$ and decode them into information packets with decoding matrices. Note that in this paper, if some packets are lost on input links to an intermediate node, the intermediate node does not forward the coded packet and discards the other received packets, even though an NC packet composed of those can be utilized in retrieving original packets. Therefore, packet loss probability shown in section VI corresponds to the worst-case performance of the proposed system.

As will be explained in section IV, all destination nodes must have information of the parity matrix for RSE coding and a coding matrix for LNC. Note that the parity matrix is determined uniquely if $(\ ,K)$ and order m of finite field **GF**(2^m) are given. Therefore, we assume that all destination nodes have the parity matrix information before the source node transmits packets.

IV. Coding Mechanisms

A. Reed Solomon Erasure Coding [13]

Suppose that the payload lengths of information packets are fixed to a multiple of integer m $(m \geq 1)$. In this case, without loss of generality, an information packet can be represented as an element of finite field **GF**(2^m) by separating the payload into symbols with length of m bits [3]. Therefore, for simplicity in description, we assume that all packets have payload with length of m bits in the rest of this section. In what follows, addition and multiplication in all coding operations are defined on **GF**(2^m).

When RSE coding is used, information packets $\boldsymbol{x} = (x_1\ x_2\ \cdots\ x_K)$ are encoded into $\boldsymbol{y} = (y_1\ y_2\ \cdots\ y_N)$ $(K \leq \ )$ by a $K \times \ $ generator matrix $\boldsymbol{G}$ for RSE coding:

$$\boldsymbol{y} = \boldsymbol{x}\boldsymbol{G}. \tag{1}$$

We use shortened RSE coding based on systematic coding, where $K \leq \ < 2^m - 1$.

RSE packets $\boldsymbol{y}$ satisfy

$$\boldsymbol{y}\boldsymbol{H}^{\mathrm{T}} = \boldsymbol{0}, \tag{2}$$

where T stands for the transpose operator and $\boldsymbol{H}$ denotes the parity matrix,

$$\boldsymbol{H} = \begin{pmatrix} \alpha^{N-1} & \alpha^{N-2} & \cdots & \alpha & 1 \\ (\alpha^2)^{N-1} & (\alpha^2)^{N-2} & \cdots & \alpha^2 & 1 \\ \vdots & \vdots & & \vdots & \vdots \\ (\alpha^{N-K-1})^{N-1} & (\alpha^{N-K-1})^{N-2} & \cdots & \alpha^{N-K-1} & 1 \\ (\alpha^{N-K})^{N-1} & (\alpha^{N-K})^{N-2} & \cdots & \alpha^{N-K} & 1 \end{pmatrix}, \tag{3}$$

where α^{i-1} $(i = 1, 2, \ldots, 2^m - 1)$ denotes the i-th element of $\mathbf{GF}(2^m)$. Generally, in RSE coding, destination nodes retrieve all information packets even when $\quad - K$ or less RSE packets are lost.

B. Linear Network Coding [9], [11], [22]

When LNC is used, every source-destination pair has disjoint paths and RSE packets are transmitted from the source node on the paths. In the network, different source-destination pairs share output link of some intermediate nodes. These intermediate nodes encode received packets from different input links into a single packet. Let $\boldsymbol{p}^{\ r} \ = \ p_1^{\ r} \ p_2^{\ r} \ \cdots \ p_L^{\ r} \quad (L \leq \quad)$ denote a set of packets received at intermediate node r. A coding vector $\widetilde{\boldsymbol{c}}^{\ r} \ = \ \widetilde{c}_1^{\ r} \ \widetilde{c}_2^{\ r} \ \cdots \ \widetilde{c}_L^{\ r}$ is assigned, where components $\widetilde{c}_i^{\ r}$ $(i \ = \ 1, 2, \ldots, \quad)$ are chosen from $\mathbf{GF}(2^m)$, and $\boldsymbol{p}^{\ r}$ is encoded into a single output packet $p^{\ r}{}_{\mathrm{t}}$,

$$p^{\ r}{}_{\mathrm{t}} = \sum_{i=1}^{L} \widetilde{c}_i^{\ r} \ p_i^{\ r} \ .$$

Because input packets themselves can be encoded ones, the output packet $p^{\ r}{}_{\mathrm{t}}$ can be represented in terms of RSE packets $\boldsymbol{y} = (y_1 \ y_2 \ \ldots y_N)$:

$$p^{\ r}{}_{\mathrm{t}} = \sum_{i=1}^{N} c_i^{\ r} \ y_i, \quad c_i^{\ r} \ \in \mathbf{GF}(2^m).$$

Let $\boldsymbol{z} = (z_1 \ z_2 \ \cdots \ z_N)$ denote a set of NC packets received at a destination and $\boldsymbol{c}_i = (c_{i,1} \ c_{i,2} \ \cdots c_{i,N})$ $(c_{i,j} \in \mathbf{GF}(2^m))$ denote the coding vector associated with z_i $(i = 1, 2, \ldots, \quad)$. $\boldsymbol{z}$ can be represented with coding matrix $\boldsymbol{C}$:

$$\boldsymbol{z} = \boldsymbol{y}\boldsymbol{C}^{\mathrm{T}}, \tag{4}$$

where

$$\boldsymbol{C} = \begin{pmatrix} \boldsymbol{c}_1 \\ \boldsymbol{c}_2 \\ \vdots \\ \boldsymbol{c}_N \end{pmatrix} = \begin{pmatrix} c_{1,1} & c_{1,2} & \cdots & c_{1,N} \\ c_{2,1} & c_{2,2} & \cdots & c_{2,N} \\ \vdots & \vdots & \ddots & \vdots \\ c_{N,1} & c_{N,2} & \cdots & c_{N,N} \end{pmatrix}.$$

C. Reed Solomon/Network Joint Coding

As explained in the preceding two sections, both RSE coding and LNC are linear coding. Therefore, if their encoders are defined on the same finite field, a joint decoder for RSE coding and LNC can be constructed. From (1) and (4), NC packets $\boldsymbol{z} = (z_1 \ z_2 \ \cdots \ z_N)$ are written to be

$$\boldsymbol{z} = \boldsymbol{x}\boldsymbol{G}\boldsymbol{C}^{\mathrm{T}}. \tag{5}$$

On the other hand, from (1) and (2), the following equation is obtained:

$$\boldsymbol{x}\boldsymbol{G}\boldsymbol{H}^{\mathrm{T}} = \boldsymbol{0}. \tag{6}$$

Note that (5) and (6) represent and $\quad - K$ equations in terms of $\boldsymbol{x}$, respectively. Thus each destination node has $2 \quad - K$ equations to retrieve K information packets $\boldsymbol{x}$ if it receives NC packets. Because $\boldsymbol{G}\boldsymbol{H}^T$ in (6) is determined uniquely, destination nodes have $\quad - K$ equations before receiving NC packets. Therefore, a destination node can retrieve $\boldsymbol{x}$ if it receives any K NC packets out of ones. This indicates that the joint coding has the same erasure correction capability as RSE coding.

The decoder for the joint coding is based on the above idea. Let $z_{\ i}$ $(i = 1, 2, \ldots, K)$ denote the i-th NC packet received at a destination node, where $l_i \in \{1, 2, \ldots, \quad\}$. The destination node constructs a system of linear simultaneous equations:

$$\boldsymbol{y}\boldsymbol{A}^T = (\underbrace{0, \ldots, 0}_{N-K}, z_{\ 1}, \ldots, z_{\ K}), \tag{7}$$

where $\boldsymbol{A}$ denotes an $\quad \times \quad$ *decoding matrix* whose (i, j)-th $(i, j = 1, 2, \ldots, \quad)$ element $a_{i,j}$ is given by for $j = 1, 2, \ldots, \quad$,

$$a_{i,j} = \begin{matrix} \alpha^{i \ N-1-j} \ , & i = 1, 2, \ldots, \quad - K, \\ c_{\ i,j}, \quad i = & \quad - K + 1, \quad - K + 2, \ldots, \quad . \end{matrix}$$

Note that the first $\quad - K$ rows in $\boldsymbol{A}$ are given by the parity matrix in (3), and the rest K rows are given by coding vectors $\boldsymbol{c}_{\ i}$ $(i = 1, 2, \ldots, K)$ associated with $z_{\ i}$. The destination node retrieves $\boldsymbol{y}$ by solving (7). When *Gauss-Jordan elimination* algorithm is used, $O(\quad ^3)$ operations for multiplications and additions are required to solve the system of linear simultaneous equations [18]. Because information packets $\boldsymbol{x}$ are encoded with systematic coding, they are retrieved as the first K packets of $\boldsymbol{y}$.

We summarize the characteristics of the joint coding as follows.

- It can retrieve all information packets even when $(\quad - K)$ or less packets are lost.
- Each destination node is notified of $\quad$, K, m, and a primitive polynomial to construct $\mathbf{GF}(2^m)$ when it attends a multicast session. Because these parameters determine the parity matrix uniquely, all destination nodes have the same parity matrix information before receiving packets. Therefore, as soon as each destination node receives K NC packets, it can decode them.
- In terms of bandwidth consumption, it is more efficient than RSE coding without network coding, because LNC can reduce the number of relayed packets.

V. ROUTING ALGORITHM

The routing algorithm used in this paper establishes vertex disjoint multiple paths from the source node to every destination node. As a result, each intermediate node may receive packets destined for different destination nodes. When network coding is used, those packets are encoded into a single NC packets and it is transmitted on output links shared by different source-destination pairs. Therefore, as the number of shared links increases, network coding reduces the number of relayed packets in the network. The following routing algorithm aims at increasing the number of shared links.

Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ denote a directed graph, where $\mathcal{V} = \{v_i;\ i = 0, 1, \ldots |\mathcal{V}| - 1\}$ represents the set of all nodes and $\mathcal{E} = \{(u, v);\ u, v \in \mathcal{V}\}$ does the set of links. Let $\mathcal{V}_{\mathrm{R}} \subset \mathcal{V}$ denote a set of destination nodes. Without loss of generality, v represents the source node. We define $\boldsymbol{W}$ as a $|\mathcal{V}| \times |\mathcal{V}|$ matrix given by

$$\begin{aligned} \boldsymbol{W} &= \{w_{i,j}\},\ 0 \le i, j \le |\mathcal{V}| - 1, \\ w_{i,j} &= \begin{matrix} 1, & \text{if } (v_i, v_j) \in \mathcal{E}, \\ \infty, & \text{otherwise.} \end{matrix} \end{aligned}$$

We set weight ϵ ($0 < \epsilon \le 1$) for updating cost on disjoint paths. Let $\mathcal{T}_j^{\ r_i}$ $(i = 1, 2, \ldots, |\mathcal{V}_R|, j = 1, 2, \ldots, \quad)$ denote a set of links consisting of the j-th disjoint paths from the source node v to a destination node r_i

Algorithm 1 shows our routing algorithm to establish vertex disjoint paths from the source node to each destination node. In the algorithm, $\mathit{DisjointPath}\,(\boldsymbol{W}, \quad, v \ , r_i)$ denotes a procedure given in [2], which is described in appendix A.

Algorithm 1: Vertex disjoint multiple paths algorithm for network coding.

```
Input  : $\boldsymbol{W}$,  , $v$ , $\mathcal{V}_{\mathrm{R}} = \{r_1, r_2, \ldots, r_{|\mathcal{V}_{\mathrm{R}}|}\}$, $\epsilon$
Output: $\mathcal{T}_j^{\ r_i}$  $(i = 1, 2, \ldots, |\mathcal{V}_{\mathrm{R}}|, j = 1, 2, \ldots, \quad)$
begin
  step 1: $i \leftarrow 1$
  step 2: Perform $\mathit{DisjointPath}\,(\boldsymbol{W}, \quad, v \ , r_i)$ and
          establish vertex disjoint   paths from $v$
          to $r_i$.
  step 3: Set the established vertex disjoint paths to be
          $\mathcal{T}_j^{\ r_i}$  $(j = 1, 2, \ldots, \quad)$.
  step 4: For all $(v_\alpha, v_\beta) \in \mathcal{T}_j^{\ r_i}$  $(0 \le \alpha, \beta \le |\mathcal{V}| - 1)$,
          set $w_{\alpha,\beta} = w_{\beta,\alpha} = \epsilon$, .
  step 5: $i \leftarrow i + 1$. If $i \le |\mathcal{V}_{\mathrm{R}}|$, go to step 2.
end
```

In step 4, $\boldsymbol{W}$ is updated by setting $w_{\alpha,\beta} = \epsilon$. This means that links on already established paths $\mathcal{T}_j^{\ r_l}$ $(l = 1, 2, \ldots, i-1, j = 1, 2, \ldots, \quad)$ are likely to be used by disjoint paths from v to r_i. By doing so, we expect the increase of the number of intermediate nodes that perform network coding.

VI. Performance Evaluation

A. Simulation Experiments

In this section, we evaluate the proposed system with simulation experiments. We use our original simulator in C++ language. Fig. 2 shows the network topology for the simulation experiments. We use a static wireless multihop network, where there are 49 nodes placed on vertices of 7 × 7 rectangular grid. The source node is placed at $(0, 0)$ and destination nodes are chosen randomly. All nodes within communication range $R = 40$ [m] can communicate with each other directly. Packets are lost randomly on links of their paths. Let $p_{\ \mathrm{ink}}$ denote the packet loss probability on each link. In this paper, we set $p_{\ \mathrm{ink}} = 0.01$. Note that the order m of finite field $\mathbf{GF}(2^m)$ affects the maximum number of users which can be accommodated in the network [5]. In this paper, we assume that m is large enough to support the number $_{\mathrm{d\ t}}$ of destination nodes.

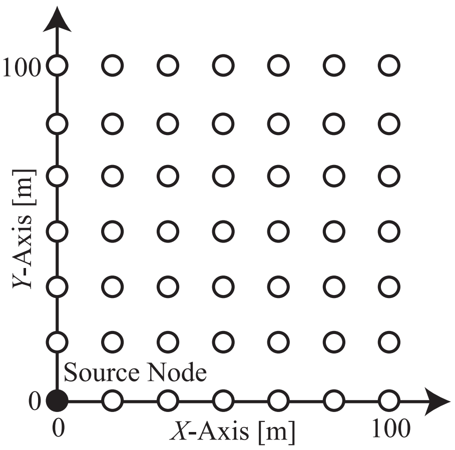

Fig. 2. Network topology for simulation experiments.

We compare the performance of the proposed system with that of RSE coding without LNC, which we refer to as *RSE source coding*. In RSE source coding, RSE packets are transmitted on a shortest-path multicast tree.

1) Network Resource Usage: We first evaluate the performance of the routing algorithm. We define the network resource usage $_{\mathrm{pkt}}(\quad, \ _{\mathrm{d\ t}}, \epsilon)$ as the number of packets transmitted by the source node and intermediate nodes every K information packets. $_{\mathrm{pkt}}(\quad, \ _{\mathrm{d\ t}}, \epsilon)$ is a function of , the number $_{\mathrm{d\ t}}$ of destination nodes, and ϵ. We also define the normalized network resource usage γ as

$$\gamma = \frac{\overline{_{\mathrm{pkt}}(\quad, \ _{\mathrm{d\ t}}, \epsilon)}}{\overline{_{\mathrm{pkt}}(1, \ _{\mathrm{d\ t}}, 1)}},$$

where $\overline{_{\mathrm{pkt}}(\quad, \ _{\mathrm{d\ t}}, \epsilon)}$ represent the average of $_{\mathrm{pkt}}(\quad, \ _{\mathrm{d\ t}}, \epsilon)$. Note that $_{\mathrm{pkt}}(1, \ _{\mathrm{d\ t}}, 1)$ corresponds to the network resource usage of RSE source coding, because it uses the shortest-path multicast tree.

Fig. 3 shows γ vs. ϵ for $_{\mathrm{d\ t}} = 10$, where γ is calculated by generating 1000 sets of 10 destination nodes which are chosen randomly from all nodes except the source node. From the figure, we observe that γ is smaller than 1 because LNC aggregates several input links on established paths into a single output link. We also observe that the effect of LNC is significant as ϵ decreases and the network resource usage is reduced more than 20% when $\epsilon \le 0.8$. The reason is as follows. When $\epsilon = 1$, the routing algorithm establishes optimal vertex disjoint paths for each destination node in terms of the number of hops. On the other hand, when $\epsilon < 1$, although established paths are not necessarily optimal, the number of links used by all destination nodes is reduced because some links are shared among paths for different destination nodes. Although γ is an increasing function of ϵ for $\epsilon > 0.4$, γ for $\epsilon = 0.2$ is slightly larger than γ for $\epsilon = 0.4$,

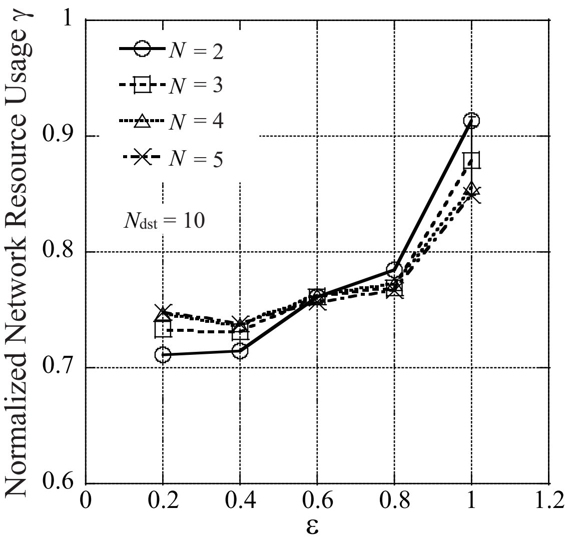

Fig. 3. Normalized network resource usage γ vs. ϵ.

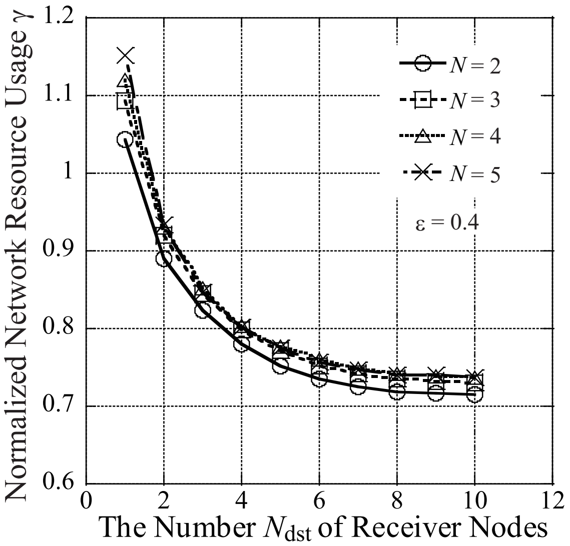

Fig. 4. Normalized network resource usage γ vs. the number N_{dst} of destination nodes.

which means that the network resource usage is a convex function of ϵ. The reason is that the number of hops increases as ϵ decreases. In what follows, we use $\epsilon = 0.4$.

Fig. 4 shows γ vs. $_{\mathrm{d\ t}}$. We observe that γ decreases as $_{\mathrm{d\ t}}$ increases because of the effect of LNC. Note that we observe $\gamma > 1$ for $_{\mathrm{d\ t}} = 1$, which means that the proposed coding mechanism consumes larger network resource usage than the shortest-path based routing algorithm. When $_{\mathrm{d\ t}} = 1$, there is no effect of network coding and the routing algorithm increases the network resource usage because it uses disjoint multiple paths instead of the shortest path. When $_{\mathrm{d\ t}} \geq 2$, however, we observe $\gamma < 1$ because of the effect of network coding. We also observe that γ is more reduced for larger $_{\mathrm{d\ t}}$ because the number of links shared among paths increases with $_{\mathrm{d\ t}}$.

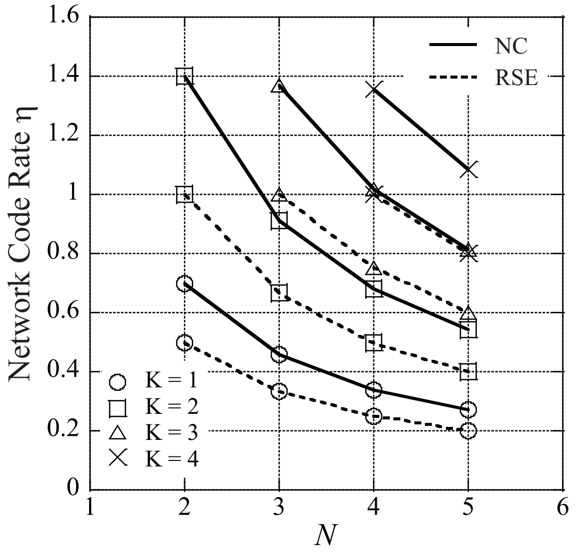

Fig. 5. Network code rate η vs. N for N_{dst} .

2) Packet Loss Probability: We evaluate packet loss probability of the proposed system by comparing with that of RSE source coding. We generate 50 sets of $_{\mathrm{d\ t}}$ destination nodes, which are located randomly on the rectangular grid topology in Fig. 2. For each set, 10^6 sets of RSE source packets are transmitted from the source node and we calculate the packet loss probability P defined as

$$P \quad = 1 - \frac{1}{_{\mathrm{d\ t}}} \sum_{i=1}^{N_{\mathrm{dst}}} \frac{^{i}_{\mathrm{rx}}}{_{\mathrm{tx}}},$$

where $_{\mathrm{tx}}$ and $^{i}_{\mathrm{rx}}$ denote the number of information packet generated at the source node and the number of retrieved information packets at i-th destination node ($i = 1, 2, \ldots,$ $_{\mathrm{d\ t}}$), respectively.

In order to evaluate the erasure correction capability and the reduction of the network resource usage in the proposed system comprehensively, we define network code rate η as

$$\eta = \frac{K}{\gamma}.$$

Smaller network code rate means that more packets are transmitted into the network. When RSE source coding is used, $\eta = K/$ because $\gamma = 1$. Fig. 5 shows network code rate η vs. for different K. In the following figures, “NC” and “RSE” stand for the proposed system and RSE source coding, respectively. We observe that the proposed coding mechanism gives higher network code rate than RSE source coding.

Fig. 6 shows P of the proposed system and RSE coding vs. the number $_{\mathrm{d\ t}}$ of destination nodes. We observe that P does not depend on $_{\mathrm{d\ t}}$. We also observe that P of the proposed system is slightly higher than that of RSE source coding. The reason is that intermediate nodes discard packets when they do not receive packets from some input links as explained in section III. However, considering the network resource

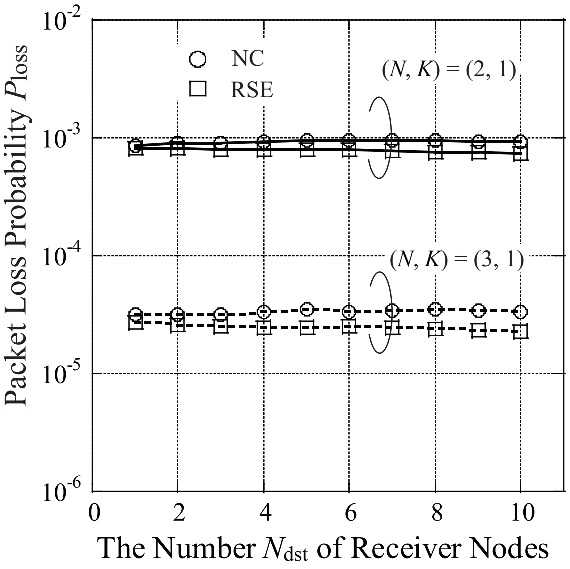

Fig. 6. Packet loss probability $_{\rm loss}$ vs. the number $N_{\rm dst}$ of destination nodes.

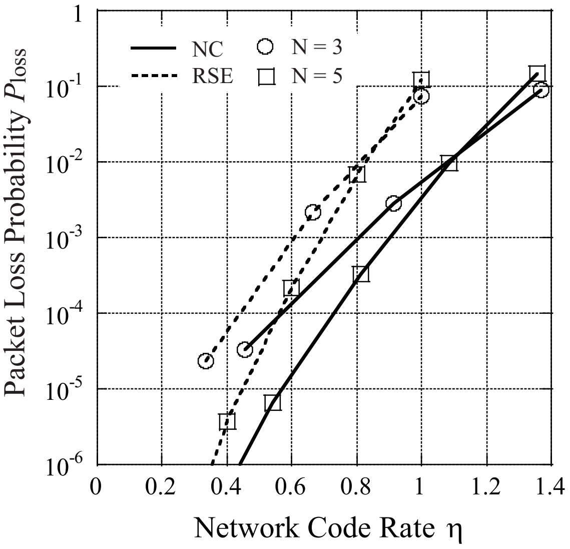

Fig. 7. Packet loss probability $_{\rm loss}$ vs. network code rate η for the proposed coding mechanism and RSE source coding for $N_{\rm dst}$.

usage, the proposed system provides a higher erasure correction capability, as shown in Fig. 7, where P vs. network code rate η are plotted. When $= 5$, P of the proposed system is lower than that of RSE source coding by a factor of 10.

VII. Concluding Remarks

In this paper, we proposed a new multicast system with network coding. In the proposed system, information packets are encoded with RSE coding at source nodes and RSE encoded packets are encoded with LNC at intermediate nodes. The proposed system is highly robust and efficient because RSE coding enhances robustness against packet erasures and LNC reduces network resource usage.

We have evaluated the network resource usage and packet loss probability in order to clarify the erasure correction capability of the proposed system. The proposed system provides higher packet erasure correction capability than RSE coding without LNC. As future research topics, we will evaluate application specific characteristics of the proposed system such as delay performance in streaming services and the feedback implosion problem in reliable file transfer services. Although the routing algorithm used in this paper is an extension of a link-state routing algorithm for vertex disjoint paths [2], we will also study routing algorithms appropriate to the joint coding.

In this paper, we have applied the joint coding to the static network environment that valid coding vectors are assigned to all intermediate nodes. The joint coding, however, is also appropriate to dynamic network environments with Randomized Network Coding (RNC) [8], where coding vectors are assigned randomly to the intermediate nodes and failure of decoding may occur due to invalid coding vectors. When RNC is used in the joint coding, destination nodes retrieve information packets from several decoding matrices. Suppose NC packets are lost and a destination node receives only $-$ NC packets, where we assume $0 \leq \quad \leq \quad - K$. The number $\kappa(\ , K, \quad)$ of decoding matrices that we can construct is given by

$$\kappa(\ , K, \quad) = \quad \frac{-}{K} \quad .$$

Information packets can be retrieved if one of those decoding matrices is nonsingular.

Appendix A
Vertex Disjoint Multiple Paths Algorithm [2]

$DisjointPath(\boldsymbol{W}, \ , \ , r)$ in algorithm 1 is a procedure to establish vertex disjoint paths from source node to destination node r. In general, vertex disjoint paths (≥ 2) are obtained from a knowledge of vertex disjoint $- 1$ paths and disjoint (> 1) paths are obtained recursively from $= 1$. Assume that vertex disjoint $- 1$ paths are established. Associated with $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, consider the *modified graph* $\tilde{\mathcal{G}} = (\tilde{\mathcal{V}}, \tilde{\mathcal{E}})$ with $2\,|\mathcal{V}|$ nodes, where $\widetilde{\mathcal{V}} = \{\tilde{v}_i;\ i = 0, 1, \ldots, 2\,|\mathcal{V}| - 1\}$, $\widetilde{\mathcal{E}} = \{(\tilde{u}, \tilde{v});\ \tilde{u}, \tilde{v} \in \widetilde{\mathcal{V}}\}$. Mapping $\mathcal{E} \rightarrow \widetilde{\mathcal{E}}$ is bijective; $(\tilde{v}_{2k}, \tilde{v}_2 \ ) \in \widetilde{\mathcal{E}}$ if and only if $(v_k, v \ ) \in \mathcal{E}$. Let denote $\boldsymbol{W} = \{\tilde{w}_{i,j}\}$ $(0 \leq i,\ j \leq 2\,|\mathcal{V}| - 1)$ to be a $2\,|\mathcal{V}| \times 2\,|\mathcal{V}|$ matrix, where

$$\begin{aligned} \tilde{w}_{2k,2} &= w_{k, } \ , \\ \tilde{w}_{2k \quad 1,2 \quad 1} &= \tilde{w}_{2k,2 \quad 1} = \tilde{w}_{2k \quad 1,2} = \infty, \\ (k, l &= 0, 1, \ldots, |\mathcal{V}| - 1.) \end{aligned}$$

The procedure to establish vertex disjoint paths from to r from a knowledge of vertex disjoint $- 1$ paths is as follows.

1) For all $(v_k, v\ ) \in \mathcal{T}_n{}^{\ r}$ $(n = 1, 2, \ldots, \quad - 1)$, set

$$\begin{aligned}\tilde{w}_{2\ ,2k} &= \infty\\ \tilde{w}_{2k,2} &= \infty,\\ \tilde{w}_{2k\ \ 1,2k} &= 0,\\ \tilde{w}_{2\ ,2k\ \ 1} &= -w_{k,\ },\\ \tilde{w}_{2k\ \ 1,2} &= \infty.\end{aligned}$$

2) Define $\widetilde{\mathcal{H}} = \{\tilde{v}_{2i} \in \widetilde{\mathcal{V}};\ \exists \tilde{v}_{2j} \in \widetilde{\mathcal{V}},\ (v_i, v_j) \in \mathcal{T}_n{}^{\ r}\ (n = 1, 2, \ldots, \quad - 1)\}$. For $v_{2k} \in \widetilde{\mathcal{H}}$ and $v_{2\ } \in \widetilde{\mathcal{V}} \setminus \widetilde{\mathcal{H}}$, set

$$\begin{aligned}\tilde{w}_{2k,2} &= \infty,\\ \tilde{w}_{2k\ \ 1,2} &= w_{k,\ },\\ \tilde{w}_{2\ ,2k} &= w_{k,\ }.\end{aligned}$$

3) Run $\mathit{modifiedDijkstra}(\widetilde{\boldsymbol{W}},\ \ , r)$ (see Appendix B) and obtain one path from to r. Let $\widetilde{\mathcal{A}} \subseteq \widetilde{\mathcal{E}}$ denote the set of links on the path.

4) Let $\mathcal{Z} \in \mathcal{E}$ denote the set of links contained in the established vertex disjoint   paths.
   a) Set $\mathcal{Z} = \{\phi\}$.
   b) Add links of the already established vertex disjoint $(\quad - 1)$ paths to $\mathcal{Z}$; for all $(v_k, v\ ) \in \mathcal{T}_n{}^{\ r}$ $(n = 1, 2, \ldots, \quad - 1)$, set $\mathcal{Z} = \mathcal{Z} \cup \{(v_k, v\ )\}$.
   c) Set $\mathcal{Z} = \mathcal{Z} \cup \{(v_i, v_j) \in \mathcal{E} \setminus \bigcup_{n=1}^{N-1} \mathcal{T}_n{}^{\ r};\ (\tilde{v}_{2i}, \tilde{v}_{2j}) \in \widetilde{\mathcal{A}},\}$.
   d) Define $\widetilde{\mathcal{H}}' = \{\tilde{v}_{2i\ \ 1} \in \widetilde{\mathcal{V}};\ \exists \tilde{v}_{2j} \in \widetilde{\mathcal{V}},\ (v_i, v_j) \in \mathcal{T}_n{}^{\ r}\ (n = 1, 2, \ldots, \quad - 1)\}$. Set $\mathcal{Z} = \mathcal{Z} \cup \mathcal{B} \cup \mathcal{C}$, where $\mathcal{B} = \{(v_k, v\ ) \in \mathcal{E};\ (\tilde{v}_{2k}, \tilde{v}_{2\ }) \in \widetilde{\mathcal{A}}, \tilde{v}_{2k} \in \widetilde{\mathcal{V}} \setminus \{\widetilde{\mathcal{H}} \cup \widetilde{\mathcal{H}}'\}, \tilde{v}_{2\ } \in \widetilde{\mathcal{H}}\}$ and $\mathcal{C} = \{(v_k, v\ ) \in \mathcal{E};\ (\tilde{v}_{2k\ \ 1}, \tilde{v}_{2\ }) \in \widetilde{\mathcal{A}}, \tilde{v}_{2k\ \ 1} \in \widetilde{\mathcal{H}}', \tilde{v}_{2\ } \in \widetilde{\mathcal{V}} \setminus \{\widetilde{\mathcal{H}} \cup \widetilde{\mathcal{H}}'\}\}$.
   e) If links on the path obtained in step 3 are interlaced with the already established disjoint $(\quad - 1)$ paths, remove the interlacing links; set $\mathcal{Z} = \mathcal{Z} \setminus \mathcal{D}$, where $\mathcal{D} = \{(v_k, v\ ) \in \mathcal{T}_n{}^{\ r}\ (n = 1, 2, \ldots, \quad - 1);\ (\tilde{v}_{2\ }, \tilde{v}_{2k\ \ 1}) \in \widetilde{\mathcal{A}}, \tilde{v}_{2\ } \in \widetilde{\mathcal{H}}, \tilde{v}_{2k\ \ 1} \in \widetilde{\mathcal{H}}'\}$.

   Links in $\mathcal{Z}$ compose vertex disjoint   paths.

Appendix B
Modified Dijkstra Algorithm [2]

The modified Dijkstra algorithm is a slight variant of the Dijkstra algorithm to calculate a path in a graph composed of some links with negative cost. For graphs without negative link cost, it reduces to the standard Dijkstra algorithm.

We define variables and functions used in the modified algorithm as follows.

- $\mathit{dist}(v)$: the cost of path from source node to node v.
- $\mathit{pred}(v)$: the predecessor of node v on the shortest path.
- Γ_v: the set of neighboring nodes of node v.
- $\tilde{w}_{u,v}$: the cost of link (u, v) from node u to node v.
- $\mathcal{V}$: the set of all nodes.

Algorithm 2 shows the modified Dijkstra algorithm. The shortest path from source node to destination node r can be traced back from $\mathrm{pred}(r)$. Note that if the modified graph has a negative cycle, where the sum of cost on the cycle is negative, the modified Dijkstra algorithm does not convergence. However, it was proved that there is no negative cycles in the modified graph, because links set to be negative in the -th iteration lie over the shortest path of the modified graph in the $(\quad - 1)$-st iteration (see chapter 3 in [2]).

Algorithm 2: The modified Dijkstra algorithm $\mathit{modifiedDijkstra}(\widetilde{\boldsymbol{W}},\ \ , r)$. The shortest path can be traced back from $\mathit{pred}(r)$.

```
Input  : $\widetilde{\boldsymbol{W}} = \{\tilde{w}_{i,j}\}$ $(0 \le i,\ j \le 2|\mathcal{V}| - 1)$,
         source node  , destination node $r$.
Output: the shortest path from   to  .
begin
  step 1: $\mathit{dist}(\ ) = 0$,
          $\mathit{dist}(i) = \tilde{w}_{s,i}$ if $i \in \Gamma_s$,
          $\mathit{dist}(i) = \infty$ otherwise.
          $\mathcal{Q} = \mathcal{V} \setminus \{\ \}$
          $\mathit{pred}(i) = \ $ , for $\forall i \in$
  step 2: $\mathcal{Q} = \{\phi\}$ the algorithm is failed. END.
          Find $j \in \mathcal{Q}$ such that
          $\mathit{dist}(j) = \min_{i \in \mathcal{Q}} \{\mathit{dist}(i)\}$
          If $\mathit{dist}(j) = \infty$, the algorithm is failed. END,
          Set $\mathcal{Q} = \mathcal{Q} \setminus \{j\}$.
          If $j = r$, the shortest path is found. END.
          Go to step 3.
  step 3: $\forall i \in \Gamma_j$, if $\mathit{dist}(j) + \tilde{w}_{j,i} < \mathit{dist}(i)$, set
          $\mathit{dist}(i) = \mathit{dist}(j) + \tilde{w}_{j,i}$, $\mathit{pre}\ (i) = j$, and
          $\mathcal{Q} = \mathcal{Q} \cup \{i\}$.
          Go to step 2.
end
```

Acknowledgement

This research was supported in part by Grant-in-Aid for Young Scientists (B) of the Ministry of Education, Culture, Sports, Science and Technology (MEXT) under Grant No. 19700059, and International Communications Foundation (ICF).

References

[1] R. Ahlswede, N. Cai, S.-Y. Li, and R. Yeung, "Network information flow," *IEEE Trans. Inf. Theory*, Vol. 46, No. 4, pp. 1204–1206, July 2000.

[2] R. Bhandari, *Survivable Networks: Algorithms for Diverse Routing*, Norwell MA, Kluwer Academic Publishers, November. 1998.

[3] P. A. Chou, Y. Wu, and K. Jain, "Practical network coding," in *Proc. 41st Allerton Conf. Commun., Cont., and Comput.*, Monticello, IL, October 2003.

[4] T. Clausen and P. Jacquet, "Optimized link state routing protocol (OLSR)," IETF RFC3626, October 2003.

[5] C. Fragouli and E. Soljanin, "Information flow decomposition for network coding," *IEEE Trans. Inf. Theory*, vol. 52, no. 3, pp. 829–848, March 2008.

[6] C. Fragouli and E. Soljanin, *Network Coding Applications*, Now Publishers, January 2008.

[7] S. Jaggi, M. Langberg., S. Katti, T. Ho, D. Katabi, M. Médard, and M. Effros, "Resilient network coding in the presence of Byzantine adversaries," *IEEE Trans. Inf. Theory*, Vol. 54, No. 6, pp. 2596–2603, June 2008.

[8] T. Ho, M. Médard, R. Koetter, D. R. Karger, M. Effros, and B. Leong, "A random linear network coding approach to multicast," *IEEE Trans. Inf. Theory*, Vol. 52, No. 10, pp. 4413–4430, October 2006.

[9] R. Koetter and M. Médard, "An Algebraic approach to network coding," *IEEE/ACM Transactions on Netw.*, Vol. 11, No. 5, pp. 782–795, October 2003.

[10] R. Koetter and F. R. Kschischang, "Coding for errors and erasures in random network coding," in *Proc. IEEE Int. Symp. Information Theory*, June 2007, pp. 791–795.

[11] S.-Y. Li, R. W. Yeung, and N. Cai, "Linear network coding," *IEEE Trans. Inf. Theory*, Vol. 29, No. 2, pp. 371–381, February 2003.

[12] D. S. Lun, N. Ramakar, R. Koetter, M. Médard, E. Ahmed, and L. Hyunjoo, "Achieving minimum-cost multicast: a decentralized approach based on network coding," in *Proc. INFOCOM 2005*, Vol. 3, pp. 1607–1617, March 2005.

[13] F. J. Macillams and N. J. A. Sloane, *The Theory of Error Correcting Codes*, North-Holland, Amsterdam, 1977.

[14] T. Matsuda, T. Noguchi, and T. Takine, "Broadcasting with randomized network coding in dense wireless ad hoc networks," *IEICE Trans. Commun.*, Vol. E91-B, No. 10, pp. 3216–3225, October 2008.

[15] A. J. McAuley, "Reliable broadband communications using a burst erasure correcting code," in *Proc. ACM SIGCOMM '90*, September 1990, pp. 297–306.

[16] T. Noguchi, T. Matsuda, and M. Yamamoto, "Performance evaluation of new multicast architecture with network coding," *IEICE Trans. Commun.*, Vol. E86-B, No. 6, pp. 1788–1795, June 2003.

[17] J. Nonnenmacher, E. Biersack, and D. Towsley, "Parity-based loss recovery for reliable multicast transmission," in *Proc. ACM SIGCOMM'97*, September 1997, pp. 289–300.

[18] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipies: the art of scientific computation, 3rd edition*, Cambridge Uniersity Press, 2007.

[19] S. Y. E. Rouayheb, A. Sprintson, and C. Georghiades, "Simple network codes for instantaneous recovery from edge failures in unicast connections," in *Proc. the UCSD Workshop Inf. Theory and its Applications*, February 2006.

[20] D. Silva, F. R. Kschischang, and R. Koetter, "A rank-metric approach to error control in random network coding," in *Proc. 2007 IEEE Inf. Theory Workshop Inf. Theory for Wireless Networks.*, July 2007, pp. 1–5.

[21] Y. C. Wu and P. Kung, "Minimum-energy multicast in mobile ad hoc networks using network coding," *IEEE Trans. on Commun.*, Vol. 53, No. 11, pp. 1906–1918, November 2005.

[22] R. W. Yeung, *Information Theory and Network Coding*, Springer, 2008.

[23] J. Zhang, K. B. Letaief, and P. Fan, "A distributed product coding approach for robust network coding," in *Proc. IEEE ICC 2008*, pp. 176–180, May 2008.

[24] Y. Zhu, B. Li, and J. Guo, "Multicast with network coding in application-layer overlay networks," *IEEE J. Sel. Areas Commun.*, Vol. 22, No. 1, pp. 107–120, January 2004.

Takahiro Matsuda received his B.E. with honors, M.E., and Ph.D. in communications engineering from Osaka University in 1996, 1997, 1999, respectively. He joined the Department of Communications Engineering at the Graduate School of Engineering, Osaka University in 1999. He is currently an Associate Professor in the Department of Information and Communications Technology, Graduate School of Engineering, Osaka Univesity. His research interests include performance analysis and the design of communication networks and wireless communications. He is a member of IEEE, IEICE, and IPSJ.

Tetsuya Takine is currently a Professor at the Department of Information and Communications Technology, Graduate School of Engineering, Osaka University. He was born in Kyoto, Japan, on November 28, 1961. He received B.Eng., M.Eng. and Dr.Eng. degrees in applied mathematics and physics from Kyoto University, Kyoto, Japan, in 1984, 1986 and 1989, respectively. In April 1989, he joined the Department of Applied Mathematics and Physics, Faculty of Engineering, Kyoto University, as an Assistant Professor. Beginning in November 1991, he spent one year at the Department of Information and Computer Science, University of California, Irvine, on leave of absence from Kyoto University. In April 1994, he joined the Department of Information Systems Engineering, Faculty of Engineering, Osaka University as a Lecturer, and from December 1994 to March 1998, he was an Associate Professor in the same department. From April 1998 to May 2004, he was an Associate Professor in the Department of Applied Mathematics and Physics, Graduate School of Informatics, Kyoto University. His research interests include queueing theory, emphasizing numerical computation, and its application to performance analysis of computer and communication networks. He is now serving as an area editor of Operations Research Letters and an associate editor of Queueing Systems, Stochastic Models, and International Transactions in Operational Research. He received Telecom System Technology Award from The Telecommunications Advancement Foundation in 2003, and Best Paper Awards from ORSJ in 1997, from IEICE in 2004 and 2009, and from ISCIE in 2006. He is a fellow of ORSJ and a member of IEICE, IPSJ, ISCIE, and IEEE.