INVITED SURVEY PAPER

Survey of Network Coding and Its Applications

Takahiro MATSUDA[†a)], **Taku NOGUCHI**[††b)], *and* **Tetsuya TAKINE**[†c)], *Members*

SUMMARY This survey summarizes the state-of-the-art research on *network coding*, mainly focusing on its applications to computer networking. Network coding generalizes traditional store-and-forward routing techniques by allowing intermediate nodes in networks to encode several received packets into a single coded packet before forwarding. Network coding was proposed in 2000, and since then, it has been studied extensively in the field of computer networking. In this survey, we first summarize linear network coding and provide a taxonomy of network coding research, i.e., the network coding design problem and network coding applications. Moreover, the latter is subdivided into throughput/capacity enhancement, robustness enhancement, network tomography, and security. We then discuss the fundamental characteristics of network coding and diverse applications of network coding in details, following the above taxonomy.

key words: *network coding, design problem, throughput/capacity enhancement, robustness enhancement, network tomography, security*

1. Introduction

Ahlswede et al. [1] proposed *network coding* which allows intermediate nodes in networks to encode several received packets into a single coded packet before forwarding. Contrarily, traditional coding techniques are referred to as *source-based coding*, where only source nodes encode packets. Network coding is considered as a generalization of conventional store-and-forward routing techniques and it was originally proposed in order to achieve multicast data delivery at the maximum data transfer rate in single-source multicast networks. This feature had a great impact on the research field of information theory and research on network coding was first activated in the information theory community. For example, see a bunch of papers in the special issue [55] of IEEE Transactions of Information Theory published in 2006. Since the middle of 2000's, network coding has been studied extensively in various research areas of computer networking because of its enormous potential.

This survey summarizes the state-of-the-art research on network coding, mainly focusing on its applications to computer networking. Note that this survey excludes network coding schemes implemented in the physical layer (Katti et al. [63], Popovski and Yomo [101], Wang and Giannakis [125], Zhang et al. [145]), because these schemes utilize not only network coding but also signal processing techniques in the physical layer, which are beyond the scope of this survey.

In this survey, we target readers who have some background knowledge of computer networks. If not so, readers are recommended to read a textbook for computer networks, such as Kurose and Ross [74], Tanenbaum [118]. In the rest of this section, we first introduce existing surveys and tutorials on network coding and briefly describe linear network coding. We then provide a taxonomy of research on network coding.

1.1 Surveys and Tutorials on Network Coding

There exist excellent surveys and tutorials on network coding in the literature. We introduce them to readers who would like to start studying network coding and aim at further understanding of network coding. Tutorials and textbooks on network coding theory can be found in Fragouli and Soljanin [35], Ho and Lun [52], Yeung et al. [139], Yeung [141]. Langberg and Sprintson [77] review algorithms to construct network coding and their computational complexity, which will also be discussed in Sect. 2.

Surveys of network coding applications are also found in the literature. Dimakis et al. [24] discuss distributed storage techniques with network coding. Fragouli and Soljanin [36] overview various network coding applications. The purpose of this survey is similar to that of Fragouli and Soljanin [36] and some network applications introduced there are also discussed in this survey. This survey, however, includes many recent works and research topics that are not introduced in Fragouli and Soljanin [36]. Further, contributions of this survey is not only introducing the state-of-the-art research on network coding, but also classifying network coding applications.

Up-to-date research on network coding can be found in many journals and international conferences. In particular, we would like to mention the International Symposium on Network Coding (NETCOD) [54] and the Wireless Network Coding Conference (WiNC) [56], which are annual forums for research developments of network coding up to date.

1.2 Linear Network Coding

Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ denote a directed network, where $\mathcal{V}$ and $\mathcal{E}$

Manuscript received June 30, 2010.
Manuscript revised November 29, 2010.

[†] The authors are with the Graduate School of Engineering, Osaka University, Suita-shi, 565-0871 Japan.

[††] The author is with College of Information Science and Engineering, Ritsumeikan University, Kusatsu-shi 525-8577 Japan.

a) E-mail: matsuda@comm.eng.osaka-u.ac.jp
b) E-mail: noguchi@is.ritsumei.ac.jp
c) E-mail: takine@comm.eng.osaka-u.ac.jp

DOI: 10.1587/transcom.E94.B.698

denote the sets of nodes and links, respectively. Let $s \in \mathcal{V}$ and $\mathcal{R} \subset \mathcal{V} - \{s\}$ denote a source node and a set of receiver nodes, respectively. Ahlswede et al. [1] show that network coding achieves an upper bound on the maximum data transfer rate in multicast communication from s to $\mathcal{R}$, which will be discussed in Sect. 2.

This section explains coding operations of network coding in directed acyclic networks. Although network coding can be constructed on cyclic networks, we do not consider cyclic networks in this survey for the following reasons: (i) link delays should be considered explicitly in designing network coding, and as a result, decoding complexity is increased, and (ii) to the best of our knowledge, directed acyclic networks have been considered in almost all network coding applications. Readers who are interested in network coding in cyclic networks may refer to Fragouli and Soljanin [35], Ho and Lun [52], Yeung et al. [139], Yeung [141].

Let $\hat{\mathcal{G}} = (\hat{\mathcal{V}}, \hat{\mathcal{E}})$ denote a directed acyclic network. To simplify the formulation, we focus on a subgraph $\hat{\mathcal{G}}^{(s,r)} = (\hat{\mathcal{V}}^{(s,r)}, \hat{\mathcal{E}}^{(s,r)}) \subset \hat{\mathcal{G}}$, which is composed of nodes and links on all paths from source node $s \in \hat{\mathcal{V}}$ to a specific receiver node $r \in \hat{\mathcal{R}}$. The formulation can be extended easily to the case of general directed acyclic networks with multiple source and multiple receiver nodes (Koetter and Médard [70]).

Unless otherwise stated, we use linear network coding (Koetter and Médard [70], Li et al. [80]) on finite field $\mathbf{GF}(q)$, where q denotes the size of finite field. Most network coding applications are developed with linear network coding, although non-linear network coding has also been studied, e.g., in Dougherty et al. [26], Kosut et al. [72], Lehman and Lehaman [79], Li et al. [83], Shadbakht and Hassibi [111]. In linear network coding, packets transferred through a network are viewed as symbols in $\mathbf{GF}(q)$, and arithmetic operations of those are defined on $\mathbf{GF}(q)$. In what follows, packets generated at source nodes are referred to as *native packets*, and packets encoded at source and intermediate nodes as *coded packets*. When we do not need to distinguish native and coded packets, we simply call them *packets*.

Suppose source node s generates N native packets and send them to a set of receiver nodes. We define $\boldsymbol{x}_i = (x_{i,1}\ x_{i,2} \cdots x_{i,L})$ $(i = 1, 2, \ldots, N)$ as the ith native packet generated at node s, where L denotes the number of symbols in a packet and $x_{i,j} \in \mathbf{GF}(q)$ $(j = 1, 2, \ldots, L)$ denotes the jth symbol of the ith packet $\boldsymbol{x}_i$. We also define $\boldsymbol{p}_i^{[v]} = (p_{i,1}^{[v]}\ p_{i,2}^{[v]} \cdots p_{i,L}^{[v])})$ $(i = 1, 2, \ldots, M^{[v]})$ as the ith packet received at intermediate node $v \in \hat{\mathcal{V}}^{(s,r)}$, where $M^{[v]}$ denotes the number of packets received at node v. When $M^{[v]} \geq 2$, intermediate node v constructs a linearly coded packet $\boldsymbol{p}_{\text{out}}^{(v,u)}$ for each output link $(v, u) \in \hat{\mathcal{E}}$:

$$\boldsymbol{p}_{\text{out}}^{(v,u)} = \sum_{i=1}^{M^{[v]}} \tilde{c}_i^{(v,u)} \boldsymbol{p}_i^{[v]}, \quad \tilde{c}_j^{(v,u)} \in \mathbf{GF}(q),$$

and transmits it into output link (v, u). Because received packets $\boldsymbol{p}_i^{[v]}$ themselves are regarded as coded packets, $\boldsymbol{p}_{\text{out}}^{(v,u)}$ can be represented in terms of native packets $\boldsymbol{x}_j$ $(j = 1, 2, \ldots, N)$:

$$\boldsymbol{p}_{\text{out}}^{(v,u)} = \sum_{j=1}^{N} c_j^{(v,u)} \boldsymbol{x}_j, \quad c_j^{(v,u)} \in \mathbf{GF}(q),$$

where $\boldsymbol{c}^{(v,u)} = (c_1^{(v,u)}\ c_2^{(v,u)} \cdots c_N^{(v,u)})$ is referred to as the coding vector associated with packet $\boldsymbol{p}_{\text{out}}^{(v,u)}$.

Coded packets received at receiver node r yield a system of linear equations that should be solved so as to retrieve native packets $\boldsymbol{x}_i$ $(i = 1, 2, \ldots, N)$. Let $\boldsymbol{c}_i = (c_{i,1}\ c_{i,2} \cdots c_{i,N})$ $(i = 1, 2, \ldots, M^{[r]})$ denote coding vectors associated with coded packets $\boldsymbol{y}_i$ received at receiver node r. We now define $\boldsymbol{X}$, $\boldsymbol{Y}$, and $\boldsymbol{C}$ as native packets, coded packets received at receiver node r, and *coding matrix*, respectively.

$$\boldsymbol{X} = \begin{pmatrix} \boldsymbol{x}_1 \\ \boldsymbol{x}_2 \\ \vdots \\ \boldsymbol{x}_N \end{pmatrix}, \quad \boldsymbol{Y} = \begin{pmatrix} \boldsymbol{y}_1 \\ \boldsymbol{y}_2 \\ \vdots \\ \boldsymbol{y}_{M^{[r]}} \end{pmatrix}, \quad \boldsymbol{C} = \begin{pmatrix} \boldsymbol{c}_1 \\ \boldsymbol{c}_2 \\ \vdots \\ \boldsymbol{c}_{M^{[r]}} \end{pmatrix}. \tag{1}$$

It then follows that $\boldsymbol{Y} = \boldsymbol{C}\boldsymbol{X}$, i.e.,

$$\begin{pmatrix} \boldsymbol{y}_1 \\ \boldsymbol{y}_2 \\ \vdots \\ \boldsymbol{y}_{M^{[r]}} \end{pmatrix} = \begin{pmatrix} c_{1,1} & c_{1,2} & \cdots & c_{1,N} \\ c_{2,1} & c_{2,2} & \cdots & c_{2,N} \\ \vdots & \vdots & \ddots & \vdots \\ c_{M^{[r]},1} & c_{M^{[r]},2} & \cdots & c_{M^{[r]},N} \end{pmatrix} \begin{pmatrix} \boldsymbol{x}_1 \\ \boldsymbol{x}_2 \\ \vdots \\ \boldsymbol{x}_N \end{pmatrix}. \tag{2}$$

Note that $M^{[r]} \geq N$ and $\operatorname{rank} \boldsymbol{C} = N$ only if native packets $\boldsymbol{x}$ can be retrieved.

If no operation is considered among symbols in the same packet, Eq. (2) can be simplified by setting $L = 1$ without the loss of generality. We define $\boldsymbol{x} = (x_1\ x_2 \cdots x_N)^{\mathrm{T}}$ and $\boldsymbol{y} = (y_1\ y_2 \cdots y_{M^{[r]}})^{\mathrm{T}}$ as native packets and packets received at receiver node r, respectively, where T denotes the transpose operator. Eq. (2) is then rewritten to be $\boldsymbol{y} = \boldsymbol{C}\boldsymbol{x}$, i.e.,

$$\begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_{M^{[r]}} \end{pmatrix} = \begin{pmatrix} c_{1,1} & c_{1,2} & \cdots & c_{1,N} \\ c_{2,1} & c_{2,2} & \cdots & c_{2,N} \\ \vdots & \vdots & \ddots & \vdots \\ c_{M^{[r]},1} & c_{M^{[r]},2} & \cdots & c_{M^{[r]},N} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{pmatrix}. \tag{3}$$

The formulations based on Eqs. (2) and (3) are referred to as the *symbol-based format* and the *packet-based format*, respectively, of linear network coding. Unless otherwise stated, the packet-based format is assumed in the rest of this survey.

Figure 1 shows an example of network coding, where nodes S and R correspond to the source node and the receiver node, respectively. Each of intermediate nodes C, E, and F has two input links and one output link, and coefficients (c_1, c_2), (c_3, c_4), and (c_5, c_6) are assigned to output links of node C, E, and F, respectively. We call intermediate nodes with multiple input links *coding nodes*. On the other hand, intermediate nodes A, B, and D dedicate themselves to forwarding received packets without coding. Source node S transmits native packets x_1 and x_2 to its two

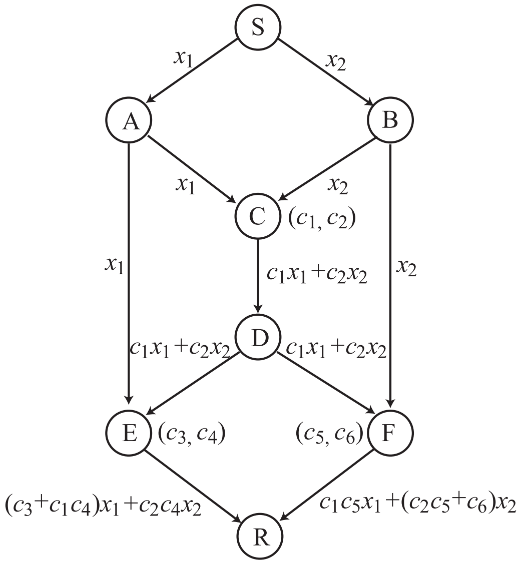


Fig. 1 An example of linear network coding.

output links (S,A) and (S,B), respectively, and receiver node R receives two coded packets $y_1 = (c_3 + c_1c_4)x_1 + c_2c_4x_2$ and $y_2 = c_1c_5x_1 + (c_2c_5 + c_6)x_2$. Therefore, the system of linear equations for native packets x_1 and x_2 is given by

$$\begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} c_3 + c_1c_4 & c_2c_4 \\ c_1c_5 & c_2c_5 + c_6 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}.$$

In order to retrieve native packets, receiver nodes need to know coding vectors of coded packets they received. Chou et al. [19] proposed a method of embedding a coding vector in the header of the coded packet in order to deliver coding vectors to receiver nodes. Assume that each native packet consists of L symbols on $\mathbf{GF}(2^m)$ and network coding is defined on $\mathbf{GF}(2^m)$, where m is called the order of finite field. Note that the payload of a packet is of fixed size $D = mL$ [bits]. In this method, the payload of a coded packet is divided into L symbols of the same size m [bits] and each symbol is taken from $\mathbf{GF}(2^m)$.

1.3 Taxonomy of Network Coding Research

This subsection presents a taxonomy of network coding research. We first classify network coding research into *network coding design problems* and *network coding applications*. Furthermore, network coding applications are subdivided into four research areas, which are summarized in Table 1.

1.3.1 Network Coding Design Problems

In network coding design problems, we have to determine paths from source nodes to receiver nodes in a given network and assign coding vectors to output links of intermediate coding nodes. Research on network coding design problems is summarized in Sect. 2. We first discuss the throughput benefit of network coding in general network topologies, and then, we consider network coding design problems in maximizing the throughput for single multicast connections and multiple unicast/multicast connections.

1.3.2 Network Coding Applications

Although network coding was originally proposed in order to maximize throughput performance in multicast communications, it has been adopted in a wide range of applications in computer networking. We classify them into *throughput/capacity enhancement*, *robustness enhancement*, *network tomography*, and *security*, depending on the purpose of utilizing network coding. Thus, in this paper, *applications* of network coding are used in a broad sense and they are divided largely into two categories: performance enhancement techniques with network coding for existing network systems and new systems based on network coding.

In the former category of applications, the performance of target network systems are enhanced with the help of network coding, although they can work without network coding. In the latter category of applications, however, network coding is an essential technology for systems. Throughput/capacity enhancement and robust enhancement techniques belong in the former category, and network tomography techniques in the latter category. Security in network coding is a newly emerging research area associated with the invention of network coding. Even though security does not exactly fit either of the above-mentioned categories, we introduce it in this survey because it will be more important when network coding is deployed widely.

(i) Throughput/Capacity Enhancement

Throughput/capacity enhancement techniques in wired and wireless network scenarios are discussed in Sect. 3. In the wired network scenario, network coding is applied to overlay networks, such as p2p (peer-to-peer) networks. On the other hand, in the wireless network scenario, it is applied to multi-hop wireless networks, such as wireless ad hoc networks, wireless sensor networks, and wireless mesh networks. Although network coding is applicable to core routers and switches in wired networks (Noguchi et al. [99], Sasaki et al. [106]), it is implemented at end hosts in these scenarios. In view of the current IP technology and protocol stacks (a set of protocols in respective layers), it seems to be difficult to implement network coding at IP core routers and switches.

(ii) Robustness Enhancement

Robustness enhancement is discussed in Sect. 4, where two techniques to improve robustness are considered: *network error correcting codes* and *network erasure correcting codes*. These coding techniques are regarded as a kind of forward error correction (FEC) based on network coding. In order to discriminate network coding-based FEC techniques from traditional FEC techniques

Table 1 Taxonomy of network coding applications.

Network Coding Applications		Application Type
Throughput/Capacity Enhancement (Sect. 3)	Overlay Networks (Sect. 3.1) · Distributed Storage (Sect. 3.1.1) · Content Distribution (Sect. 3.1.2) · Layered Multicast (Sect. 3.1.3)	Performance Enhancement with Network Coding for Existing Network Systems
	Wireless Networks (Sect. 3.2) · Throughput Enhancement (Sect. 3.2.1) – Physical Layer Approach – Data Link Layer Approach – Network Layer Approach · Broadcast Storm Problem (Sect. 3.2.2)	
Robustness Enhancement (Sect. 4)	Network Error Correcting Codes (Sect. 4.1) Network Erasure Correcting Codes (Sect. 4.2) · Network Coding-Based Routing	
Network Tomography (Sect. 5)	Link Loss Rate Inference (Sect. 5.1) Topology Inference (Sect. 5.2)	New Techniques Based on Network Coding
Security (Sect. 6)	Pollution Attacks (Sect. 6.1) Eavesdropping (Sect. 6.2)	Performance Enhancement of Network Systems with Network Coding

that only source nodes encode packets, we refer traditional error (resp. erasure) correcting codes as *source-based error* (resp. *erasure*) *correcting codes*. In order to improve robustness, source-based error and erasure correcting codes utilize temporal redundancy, that is, redundant packets or symbols are appended to native packets before transmission. On the other hand, network error and erasure correcting codes utilize spatial redundancy of network topologies; they improve robustness by using multiple paths from a source node to a receiver node and mixing native packets at intermediate nodes.

(iii) Network Tomography

Network tomography has a different purpose from other network coding applications because it aims to infer characteristics of networks on which packets are delivered. In Sect. 5, two kinds of network tomography techniques with network coding are summarized, i.e., link loss rate inference and topology inference techniques. Even though network tomography can be done without network coding, network tomography based on network coding has some advantages, such as the improvement of accuracy and the reduction of complexity in choosing which paths to monitor [31].

(iv) Security

In Sect. 6, two security issues, *pollution attacks* and *eavesdropping*, in network coding are summarized. In pollution attacks, malicious users intentionally inject corrupted packets with the aim of contaminating information to be received at receiver nodes. Network coding is particularly vulnerable to pollution attacks because several packets are mixed at intermediate nodes on the way to data transfer and they are distributed over receiver nodes. Some techniques to foil pollution attacks, which correct or remove corrupted packets, are described. On the other hand, eavesdropping means that unintended users or adversaries receive information by accessing some links on which packets are forwarded. *Secure network coding* has been proposed in order to prevent eavesdroppers from obtaining meaningful information.

Sections 2–6 discuss the above research topics in details and Sect. 7 provides the summary and concluding remarks.

2. Network Coding Design

Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ denote a directed network with positive capacity $u_{i,j}$ associated with link $(i, j) \in \mathcal{E}$. We first consider data transfer from a single source node $s \in \mathcal{V}$ to a single receiver node $r \in \mathcal{V}$ $(s \neq r)$ with multiple paths. It is well known that the maximum achievable data transfer rate from source node s to receiver node r corresponds to the magnitude $h_{s,r}$ of the maximum flow, which is equivalent to the capacity of minimum s-r cuts. Several algorithms for finding the maximum flows are available. Readers may refer to Ahuja et al. [3], Wang and Lin [124], for example.

We now consider multicast data transfer from source node $s \in \mathcal{V}$ to a set $\mathcal{R} \subset \mathcal{V} - \{s\}$ of receiver nodes, where data transfer rates to respective receivers in $\mathcal{R}$ are assumed to be identical. We define $h(s, \mathcal{R})$ as

$$h(s, \mathcal{R}) = \min_{r \in R} h_{s,r},$$

where $h_{s,r}$ $(r \in \mathcal{R})$ denotes the capacity of minimum s-r cuts. By definition, $h(s, \mathcal{R})$ gives an upper bound on the maximum data transfer rate in multicast communication from node s to receiver nodes in $\mathcal{R}$. Due to resource contention, however, the upper bound $h(s, \mathcal{R})$ cannot necessarily be achieved only by routing. Ahlswede et al. [1, Theorem 1] show that the upper bound $h(s, \mathcal{R})$ can be achieved with the help of network coding. Furthermore, Li et al. [80, Theorem 3.3] show that $h(s, \mathcal{R})$ can be achieved with linear network coding.

2.1 Throughput Benefit of Network Coding

As stated above, the achievable throughput $h(s, \mathcal{R})$ with network coding can be larger than the achievable throughput $h_R(s, \mathcal{R})$ only with routing. If so, one may ask how large

$h(s,\mathcal{R})$ is, compared with $h_R(s,\mathcal{R})$. Unfortunately, there is no general answer; it depends on the scenario. We define the *throughput benefit* of network coding as a ratio $h(s,\mathcal{R})/h_R(s,\mathcal{R})$ of the achievable throughput of network coding to that of routing. Chekuri et al. [15] discuss bounds of the throughput benefit of network coding for several regular configurations of directed acyclic networks. They show that, while there exist some configurations such that $h(s,\mathcal{R})/h_R(s,\mathcal{R}) = O(\sqrt{|\mathcal{V}|})$, it can be $h(s,\mathcal{R})/h_R(s,\mathcal{R}) \leq 1 + 1/(e-1) \approx 1.58$ for any $|\mathcal{V}|$ in some other cases. In other words, the throughput benefit of network coding can increase boundlessly with the number $|\mathcal{V}|$ of nodes in some cases, but in some other cases, the throughput benefit is limited and it is bounded by $1 + 1/(e-1)$, regardless of the number of nodes.

The throughput benefit of network coding in undirected networks is also examined in Li and Li [81], where the capacity $u_{i,j}$ of link $(i,j) \in \mathcal{E}$ represents the maximum of the sum of data transfer rates in both directions. Such a network might be useful in modeling a certain class of wireless networks. It is shown in Li and Li [81, Theorem 3] that $h(s,\mathcal{R})/h_R(s,\mathcal{R}) \leq 2$, i.e., the throughput benefit in undirected networks is bounded above by two, and in many cases, the throughput benefit is close to one. Thus the throughput benefit of network coding in undirected networks is very limited in general. See Sect. 3.2 also.

2.2 Single Multicast Connections

Suppose that a directed network $\mathcal{G} = (\mathcal{V},\mathcal{E})$, positive capacity $u_{i,j}$ associated with link $(i,j) \in \mathcal{E}$, a single source node $s \in \mathcal{V}$, and a set $\mathcal{R} \subset \mathcal{V} - \{s\}$ of receiver nodes are given. Source node s is assumed to send the same information to all receiver nodes at the same rate h, where $h \leq h(s,\mathcal{R})$. In this scenario, we have to determine a multicast network (i.e., a directed, acyclic sub-network containing only multipath routes from node s to receivers) and assign a network code to each of output links from coding nodes, where the problem of selecting the field size q is included in the latter.

The problem of finding a suitable multicast network is studied from two different viewpoints. One is to find a so-called minimal multicast network, and the other is to find a sub-network with the minimum cost. A sub-network over $G(\mathcal{V},\mathcal{E})$ is called a minimal multicast network if a removal of any link belonging to the sub-network would result in the decrease of its capacity (Langberg et al. [75]). Note that a minimal multicast network can be obtained by a simple greedy algorithm that redundant links are deleted successively. Langberg et al. [76] proposed an efficient algorithm for a subclass of this problem. On the other hand, the problem of finding a minimum-cost multicast network can be formulated as an optimization problem with linear constraints, and solution methods for various scenarios are discussed in Lun et al. [89]. The problem of finding a sub-network that maximizes the utility function of data transfer rate is studied in Wu and Kung [127].

Suppose a multicast network $\hat{\mathcal{G}} = (\hat{\mathcal{V}},\hat{\mathcal{E}})$ is given, where source node s sends the same information to all receiver nodes $r \in \hat{\mathcal{R}}$ at the same rate h. The set $\hat{\mathcal{V}}$ of nodes can be divided into three disjoint subsets: $\hat{\mathcal{V}} = \{s\} \cup \mathcal{R} \cup \hat{\mathcal{V}}_I$, where $\hat{\mathcal{V}}_I$ denotes a set of all intermediate nodes that relay information. Moreover, nodes in $\hat{\mathcal{V}}_I$ can be classified into two classes: $\hat{\mathcal{V}}_I = \hat{\mathcal{V}}_C \cup \hat{\mathcal{V}}_D$, where $\hat{\mathcal{V}}_C$ denotes a set of coding nodes (i.e., intermediate nodes with multiple input links) and $\hat{\mathcal{V}}_D$ denotes a set of intermediate nodes only with one input link. Note that a coding vector may be assigned to each of output links from coding nodes in $\hat{\mathcal{V}}_C$. On the other hand, nodes in $\hat{\mathcal{V}}_D$ fill the role of distributing the incoming information to downstream nodes, without coding. Based on this observation, the original multicast network $\hat{\mathcal{G}}$ can be transformed to a simpler logical network, which is called information flow decomposition in Fragouli and Soljanin [32].

For a given directed, acyclic multicast network $\hat{\mathcal{G}} = (\hat{\mathcal{V}},\hat{\mathcal{E}})$, finding the minimum required size q of finite field $\mathbf{GF}(q)$ is known to be NP-hard (Lehman and Lehman [79, Theorem 3.2]). When all links have unit capacity, $q \geq |\mathcal{R}|$ is sufficient for a feasible† code assignment (Jaggi et al. [58, Theorem 3]). A deterministic polynomial time algorithm for finding feasible network codes is developed in Jaggi et al. [58], Langberg et al. [76], Yang and Yang [134]. Note that randomly chosen network codes may be feasible. Ho et al. [50] examine a random linear network coding approach in depth. In particular, when $q > |\mathcal{R}|$, the probability that a random assignment of coding vectors is feasible is bounded below by $(1 - |\mathcal{R}|/q)^{\eta}$, where η denotes the number of output links to which coding vectors are assigned (He et al. [50, Theorem 2]). Note that this result also indicates the above-mentioned sufficient condition $q > |\mathcal{R}|$ for the existence of a feasible network coding.

2.3 Multiple Unicast/Multicast Connections

When there are multiple connections in a network, the network coding design problem becomes very complicated in general. According to Lehman and Lehman [79], we classify such situations by characterizing them with a four-tuple $(\alpha,\beta,\gamma,\delta)$, where

$$\alpha = \begin{cases} 1, & \text{single source,} \\ n, & \text{multiple sources,} \end{cases}$$

$$\beta = \begin{cases} 1, & \text{single receiver,} \\ m, & \text{multiple receivers,} \end{cases}$$

$$\gamma = \begin{cases} \text{I}, & \text{all sources transmit the same information,} \\ \text{D}, & \text{each source transmits disjoint information,} \\ \text{A}, & \text{otherwise,} \end{cases}$$

$$\delta = \begin{cases} \text{I}, & \text{all receivers demand all information,} \\ \text{D}, & \text{each receiver demands disjoint information,} \\ \text{A}, & \text{otherwise.} \end{cases}$$

---

† For a multicast network $\hat{\mathcal{G}} = (\hat{\mathcal{V}},\hat{\mathcal{E}})$, where source node s sends the same information to all receiver nodes $r \in \hat{\mathcal{R}}$, a network code is said to be *feasible* if there exists a decoding function for each receiver node $r \in \hat{\mathcal{R}}$ (Langberg et al. [76]).

Table 2 Problem classification (Lehman and Lehman [79]).

	α	β	γ	δ
Routing	1 or n	1	I,D, or A	I
	1 or n	m	I	D
Linear NC (polynomial)	1 or n	m	I,D, or A	I
Linear NC (NP-hard)	1 or n	m	I	A
Non-linear NC	n	m	D or A	D or A

Note that $\gamma = \text{I}$ in the case of single source ($\alpha = 1$) and $\delta = \text{I}$ in the case of single receiver ($\beta = 1$).

We assume that routing, linear network coding, and general (non-linear) network coding can be utilized in order to accommodate multiple connections simultaneously and that a scenario $(\alpha, \beta, \gamma, \delta)$ is *solvable*†. Table 2 shows the problem classification given in Lehman and Lehman [79]. In this table, "Routing" represents that scenarios are solvable without network coding, i.e., the throughput benefit $h(s, \boldsymbol{R})/h_R(s, \boldsymbol{R})$ of each connection is equal to one.

On the other hand, "Linear NC (polynomial)" represents that scenarios are solvable with linear network coding whose coding vectors can be obtained in polynomial time, "Linear NC (NP-hard)" represents that scenarios are solvable with network coding but the problem of finding a feasible coding vector is NP-hard, and "Non-linear NC" represents that scenarios may not be solvable with linear network coding and may require non-linear network coding. Note that in those three cases, there are multiple receivers ($\beta = m$) and the throughput benefits are greater than one. A necessary and sufficient condition that a specific scenario is solvable by linear network coding is given in Koetter and Médard [70, Theorem 6]. Research works on multiple connections can be found in Iwama et al. [57], Kim et al. [67], Price and Javidi [102], Yang and Yang [133].

3. Throughput/Capacity Enhancement Techniques

3.1 Overlay Networks

Overlay networks are logical networks residing in the application layer and one of the best feasible solutions to implementing network coding without modifying or replacing neither routers nor routing/MAC protocols. There are three well-known applications of network coding in overlay networks: *distributed storage systems, content distribution*, and *layered multicast*, which will be discussed in the rest of this subsection.

3.1.1 Distributed Storage Systems

Distributed storage systems (DSS) store data in a large number of storage nodes distributed over networks. Each storage node stores a piece of the original data file and each user recovers the entire original data file by collecting a sufficient number of pieces from those storage nodes. In conventional DSS, the redundancy is introduced by source-based erasure codes, where source nodes append the redundancy to enhance the reliability against failures and departures of storage nodes. When a Maximum Distance Separable (MDS) code, such as Reed-Solomon codes, is used to append the redundancy, the original data file of size M is divided into k pieces of the same size M/k, which are referred to as native pieces. With the source-based erasure code, these pieces are encoded into n ($n > k$) pieces of size M/k, which are referred to as coded pieces. Each coded piece is stored in one of storage nodes and any k coded pieces can recover the entire original data file.

In DSS, if a storage node fails or leaves the system, it is necessary to regenerate a new storage node in order to maintain the reliability level. This problem is called the *repair problem* in Dimakis et al. [23]. Consider a situation that one storage node fails and a new node is introduced to replace it. In an n-node DSS with a source-based erasure code, the new node has to download k coded pieces among from the remaining $n - 1$ nodes and regenerates the coded piece that the failed node stored. This repair operation is very inefficient because the new node has to download k coded pieces to regenerate one coded piece. The total amount of data to be downloaded by the new node is referred to as the *repair bandwidth*. In DSS with a source-based erasure code, the repair bandwidth is equal to M because the new node has to download k coded pieces of size M/k.

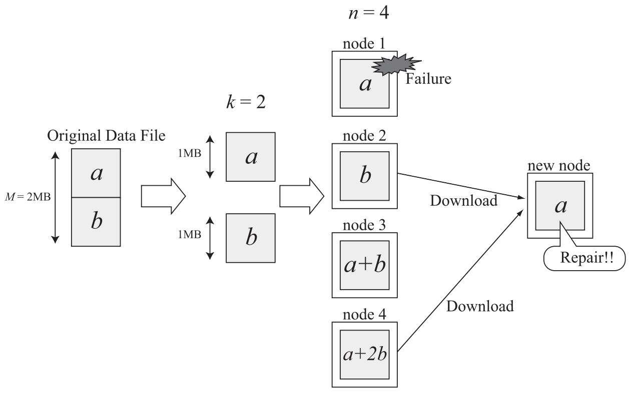
Figure 2(a) shows an example of the repair problem for a source-based erasure code, where we assume $M =$ 2 Mbytes, $k = 2$, and $n = 4$. Thus the original file is divided into two native pieces of the same size 1 Mbyte, and these native packets are encoded into four coded pieces. These coded pieces are stored in four different storage nodes: 1, 2, 3, and 4, respectively. When node 1 fails, the new node regenerates the same coded piece as that node 1 stored by downloading two coded pieces among from the remaining three nodes (from nodes 2 and 4 in this example), so that the repair bandwidth is equal to 2 Mbytes.

Dimakis et al. [23] show that network coding can help reduce the repair bandwidth if new nodes are allowed to download coded pieces from more than k nodes. Figure 2(b) shows an example of the repair problem with network coding, where each storage node stores two coded pieces of size $M/(2k) = 0.5$ Mbytes. Assume that nodes 1, 2, 3, and 4 store pieces (a_1, a_2), (b_1, b_2), $(a_1 + b_1, a_2 + b_2)$, and $(a_2 + b_1, a_1 + a_2 + b_2)$, respectively. When node 1 fails, the new node may download linear combinations $b_1 + b_2$, $a_1 + a_2 + b_1 + b_2$, and $a_1 + 2a_2 + b_1 + b_2$ from nodes 2, 3, and 4, respectively, and lost pieces (a_1, a_2) is regenerated from those downloaded pieces. As a result, the repair bandwidth is reduced to 1.5 Mbytes. The construction of network codes in DSS is discussed in Wu et al. [128], Wu [129].

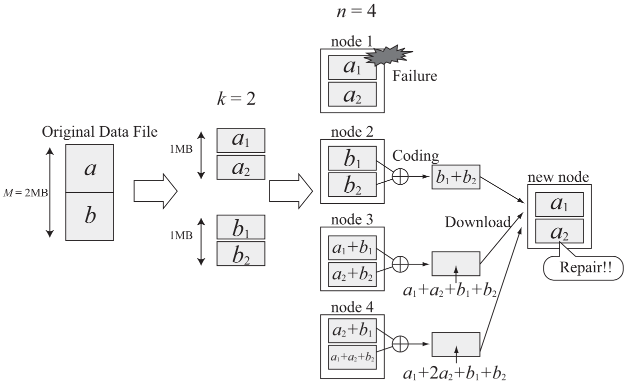
The essential and original purpose of the repair problem is not regenerating the same coded piece that the failed node stored, but regenerating a coded piece that maintains the property of MDS codes that any k pieces downloaded

---

† A scenario $(\alpha, \beta, \gamma, \delta)$ is said to be solvable if source node(s) can transmit packets to their receiver nodes at given rate(s), using either of routing, linear network coding, or general (non-linear) network coding.



(a) A distributed storage system with source-based erasure coding.



(b) A distributed storage system with network coding.

Fig. 2 An example of a repair operation.

among from n storage nodes are sufficient to recover the original data file. Dimakis et al. [24] survey research on the repair problem with network coding, where the repair problem is classified into three categories: *exact repair*, *functional repair*, and *exact repair of systematic parts*. In exact repair, the same piece that the failed node stores is regenerated. In functional repair, a coded piece different from the failed piece can be regenerated as long as the property of MDS codes is maintained. In exact repair of systematic parts, a systematic code is used to generate coded pieces, i.e., some coded pieces are the same as native pieces, which are referred to as the systematic part. Coded pieces in the systematic part are repaired exactly and coded pieces in the non-systematic part are repaired functionally. Note that systematic codes are used in both examples of Fig. 2.

3.1.2 Content Distribution

Content Distribution (CD) systems provide a mechanism to distribute contents, such as multimedia files and software, to a large number of network users. Peer-to-peer content distribution (P2PCD) systems form overlay networks with end hosts (peers) having the same interest, in order to improve the scalability and flexibility to network dynamics. P2PCD is a combination of a distributed storage function, a propagation function of contents among peers, and a request routing function. BitTorrent (Cohen [21]) is a typical example of P2PCDs.

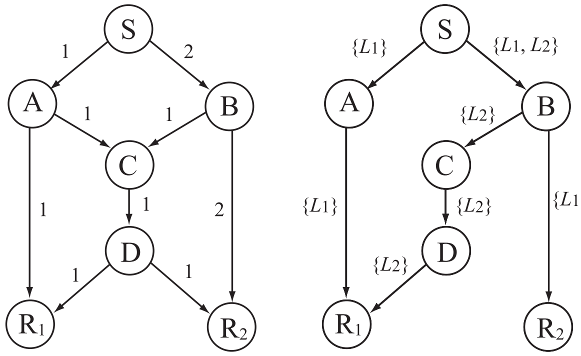
Gkantsidis and Rodriguez [39] proposed a network coding-based file sharing system, named Avalanche. In this system, an original data file is divided into n native pieces and before sending out those native pieces, each peer encodes them using random network coding. As a result, the original data file can be recovered from any n linearly independent coded pieces. Network coding has two advantages in file sharing systems. One is that file sharing systems become more robust against source/peer departures, which is the same advantage as in DSS. The other is that network coding simplifies the content propagation scheme, i.e., each peer does not need to decide which pieces to download. The benefit of network coding in P2PCD is shown in Gkantsidis and Rodriguez [39] in comparison to P2PCD systems without coding and with source-based coding. The detailed performance analysis of an Avalanche implementation is presented in Gkantsidis et al. [41], [42].

Zhang and Li [147] quantify the fundamental advantage of P2PCD with network coding in a non-cooperative environment, where peers behave selfishly and they are interested in maximizing their own payoff. The authors proposed a market model for P2PCD with network coding. In this market model, peers act as market agents who buy/sell coded pieces from/to others and strategically set prices for their possessions according to their availability in the market. The analytical result of the market model shows that P2PCD with network coding can alleviate the instability of the system due to joins and departures of peers, and it can improve the system's resilience to selfish peers who leave the system immediately after they finish downloading. Related work on network coding-based content distribution systems can be found in Chiu et al. [18], Ma et al. [90], Zhu et al. [151].

3.1.3 Layered Multicast

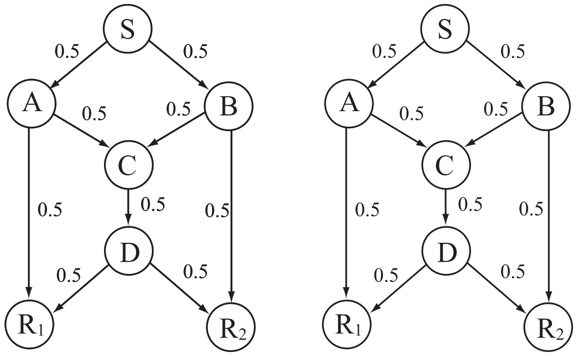
Layered multicast enables multicast data transfer to receiver nodes at different rates with the help of layered coding (Cui and Nahrstedt [22]). In layered multicast, contents to be delivered are divided into multiple layers, in each of which a separate multicast session is established. Each receiver node subscribes to the proper number of layers to maximize the throughput. Layered multicast is an effective solution to address the receiver heterogeneity in peer-to-peer overlay networks, especially for graphics or multimedia streaming. Network coding is applicable to layered multicast in order to maximize the aggregate throughput to all receivers, which is equivalent to the total number of layers subscribed by receivers when all layers are identical in transmission rate.

Figure 3 illustrates an example of the layered multicast by routing (i.e., without network coding) and those with network coding. Consider a network in Fig. 3(a), where the capacity of each link is also shown. We assume that source



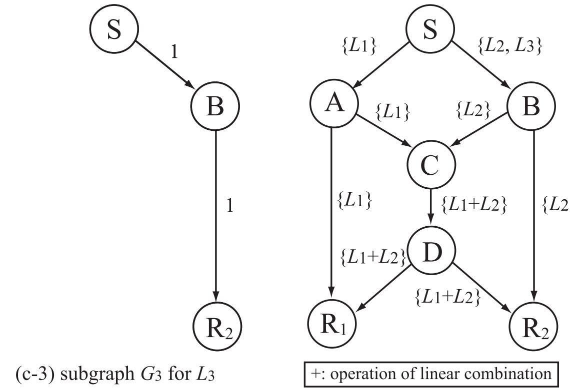
(a) Example network.

(b) Layered multicast by routing.



(c-1) subgraph G_1 for L_1 (intra-layer network coding).

(c-2) subgraph G_2 for L_2 (intra-layer network coding).



(c-3) subgraph G_3 for L_3 (intra-layer network coding).

(d) Layered multicast with inter-layer network coding.

Fig. 3 An example of layered multicast with/without network coding.

node S delivers a multimedia stream to two receiver nodes R_1 and R_2, where the multimedia stream is composed of three layers, L_1, L_2, and L_3 whose transmission rates are all equal to one. Note that the capacity of minimum S-R_1 (resp. S-R_2) cuts is equal to two (resp. three). Figure 3(b) shows an optimal layered multicast by routing, where the maximum aggregate throughput is equal to four.

On the other hand, the network in Fig. 3(a) can be split into three subgraphs G_1, G_2, and G_3, as shown in Figs. 3(c-1), (c-2), and (c-3), respectively. We assume the ith ($i = 1, 2, 3$) layer stream is transmitted with G_i. Note here that in G_i ($i = 1, 2$), network coding enables both receiver nodes R_1 and R_2 to subscribe to the ith layer L_i at rate one. Therefore the aggregate throughput is equal to five when network coding is applied in this way.

In the above example, network coding is performed within each layer, which is referred to as *intra-layer network coding*. On the other hand, Fig. 3(d) shows *inter-layer network coding*, which allows coding across layers. More specifically, intermediate node C encoded layers L_1 and L_2 into $L_1 + L_2$, i.e., packets in layers L_1 and L_2 are combined linearly. In this specific example, the achievable throughput of inter-layer network coding is the same as that of intra-layer network coding. In general, however, inter-layer network coding can achieve higher throughput than intra-layer network coding (Dumitrescu et al. [27], Kim et al. [69]).

Intra-layer network coding includes an optimization problem for finding subgraphs to achieve the maximum aggregate throughput. Chenguang et al. [17], Jinjing et al. [60], Zhao et al. [150] proposed several algorithms for intra-layer network coding. Although inter-layer network coding utilizes the full potential of network coding, it is a difficult and complicated problem, compared with intra-layer network coding. See Sect. 2.3, where inter-layer network coding is characterized by $(1, m, I, A)$ or (n, m, D, A). Dumitrescu et al. [27] proposed a layered multicast with inter-layer network coding. Kim et al. [69] proposed a simple distributed layered multicast both with intra-layer coding and with inter-layer network coding.

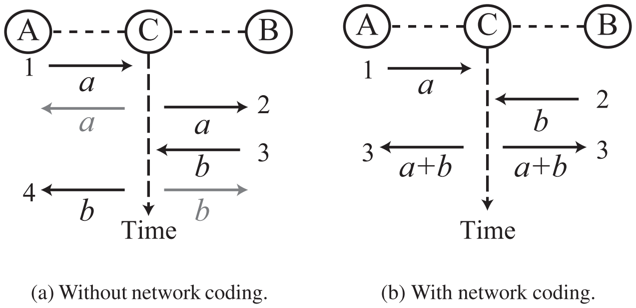
3.2 Wireless Networks

In this section, we summarize two applications of network coding to wireless networks: throughput enhancement techniques and techniques to solve the *broadcast storm problem*. In applying network coding, source nodes and some of intermediate nodes have to establish multiple links to their neighboring nodes so as to forward packets on multiple paths for each pair of source and receiver nodes. This suggests that wireless networks are suitable for network coding because wireless links inherently have the broadcast nature, i.e., signal transmitted from a wireless node may reach several neighboring nodes. Recall that in network coding, different coding vectors can be assigned to different output links. In wireless networks, however, it will be a good idea to use the same coding vector for all output links if the decrease of the number of transmissions is beneficial.

3.2.1 Throughput Enhancement

Network coding schemes for multi-hop wireless networks have been studied in order to improve their throughput performance. We explain how network coding improves the throughput performance with the following simple example.

Consider a network in Fig. 4, where node A (resp. B) delivers packet *a* (resp. *b*) to node B (resp. A) via node C. We assume that each node has an omni-directional antenna. When network coding is not used, four transmissions are required to exchange the packets as shown in Fig. 4(a). On the other hand, when network coding is applied as shown in



(a) Without network coding. (b) With network coding.

Fig. 4 Throughput enhancement with network coding (Katti et al. [64]).

Fig. 4(b), node C encodes packets *a* and *b* into packet $a + b$ and forwards it to nodes A and B. Node A (resp. B) retrieves packet *b* (resp. *a*) with coded packet $a + b$ and native packet *a* (resp. *b*). In this case, the system throughput is improved by network coding because only three transmissions are required to exchange those packets[†]. It is worth mentioning that in Fig. 4(a), node C transmits redundant packets *a* and *b* to nodes A and B, respectively. Therefore, in multi-hop wireless networks, network coding not only improves the throughput performance but also reduces redundant packet transmissions, and it results in improving energy consumption due to overhearing packets and reducing collisions of packets in random access MAC protocols.

Chachulski et al. [12] proposed an intra-session network coding, which encodes native packets generated from the same source node. On the other hand, Katti et al. [64] proposed an inter-session network coding, which encodes native packets generated from different source nodes. These two schemes have been extended by many researchers in order to increase coding opportunities by customizing several functions in different protocol layers as follows.

(i) Physical Layer Approach

In the physical layer approach, the adaptation of transmission rate is considered. In wireless networks, nodes within its communication range can communicate with each other. Increasing the communication range results in increased coding opportunities because the number of neighboring nodes increases. The communication range is considered as a function not only of the transmission power but also of the transmission rate at the node, because the decrease of the transmission rate increases the immunity of transmitted signal to noise. Also, the increase of the transmission power may cause severe interference among nodes. The transmission rate adaptation is considered in Kim and de Vaciana [68] and it was shown that an adequate rate adaptation improves the throughput performance.

(ii) Data Link Layer Approach

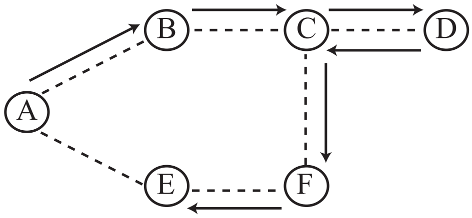
In general, network coding has the fundamental problem that coding nodes have to wait until they receive multiple packets. Therefore packet erasures and delays might cause the significant performance degradation. In order to optimize the throughput performance, MAC scheduling schemes suitable for network coding are considered in Chaporkar et al. [13], Sagduyu and Ephremides [105], Scheuermann et al. [108]. MAC scheduling is a kind of data link layer approach and it coordinates packet transmissions among nodes in order to reduce collisions. MAC scheduling without network coding decides which links should be activated and which sessions should be served on those links. When network coding is applied, MAC scheduling should further decide whether and how intermediate nodes encode packets. Umehara et al. [122] analyzed the throughput of two-hop wireless relay networks with a slotted ALOHA protocol combined with network coding. In order to improve the throughput performance, they consider an approach similar to the MAC scheduling and show that controlling the transmission probability of the relay node improves the throughput performance.

(iii) Network Layer Approach

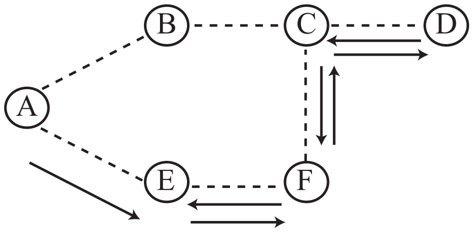
Coding-aware routing schemes are studied in Le et al. [78], Ni et al. [97], Sengupta et al. [109], [110], Zhang and Zhang [143]. The coding-aware routing is a path selection algorithm for increasing coding opportunities, so that it is a kind of network layer approach. Consider a network in Fig. 5, where nodes A and D deliver packets to nodes D and E, respectively. When the shortest path routing in terms of the number of hops is employed, nodes A and D use route A → B → C → D and route D → C → F → E, respectively. On the other hand, when the coding-aware routing is employed, node A uses route A → E → F → C → D to increase coding opportunities. In general, increasing coding opportunities results in increasing not only the number of hops but also interference among different flows because paths of these flows are established so as to use the same wireless link. Therefore the coding-aware routing is reduced to an optimization problem involving coding opportunities, the number of hops, and interference.

Keshavarz-Haddadt and Riedi [66] and Liu et al. [87] derive bounds on the throughput benefit of network coding, using the wireless network model considered in Gupta and Kumar [45]. These papers analyze the scaling behavior of wireless networks with network coding and show that at most a constant factor improvement is achievable as compared with the throughput performance without network coding (cf. Sect. 2.1). Although the throughput enhancement is one of the important applications of network coding, these results indicate that network coding could be more helpful for improving delay, reliability, and robustness, rather than improving the throughput performance, as concluded in Liu et al. [87].

[†] Celebiler and G. Stette [11] originally proposed this technique for exchanging information between two senders in 1978, where node B corresponds to a satellite repeater that uses XOR operations to generate coded packets $a + b$.



(a) Shortest path routing.



(b) Coding-aware routing.

Fig. 5 An example of coding-aware routing (Sengupta et al. [109]). Node A sends packets to node D and node D sends packets to node E.

3.2.2 Broadcast Storm Problem

Broadcasting, where a source node disseminates messages to all other nodes, is one of fundamental operations in computer networks. Although there are various methods of broadcasting (Tanenbaum [118]), flooding is a typical method in multi-hop wireless networks. Note that flooding is the simplest broadcasting method, which allows every node to forward a packet to all of the neighboring nodes when it receives the packet for the first time. In wireless networks, however, flooding may cause serious collisions of packets and/or significant energy consumption due to a huge amount of redundant packet forwarding, which is referred to as the broadcast storm problem (Ni et al. [98]). There have been several research efforts to solve the broadcast storm problem (Ni et al. [98], Peng and Lu [100], Qayyum et al. [103], Sheng et al. [112]).

Network coding is a promising technique to solve the broadcast storm problem because it can reduce the number of forwarded packets by encoding multiple packets to be forwarded into a single coded packet. Broadcasting can be classified into single-source broadcasting and multiple-source broadcasting. In the former, a single source node disseminates its packets to all other nodes, while every node disseminates its packets to all other nodes in the latter. Matsuda et al. [94] proposed a single-source broadcasting scheme with random network coding and the packet loss probability is evaluated for several parameters of network coding, such as the length of coding vectors and the order of finite field.

Multiple-source broadcasting schemes with network coding are also studied in the literature. Fragouli et al. [37] proposed distributed broadcasting algorithms for regular network topologies. Furthermore, distributed algorithms for random network topologies were proposed, where network coding is combined with a probabilistic routing algorithm that each node forwards packets with a certain probability. On the other hand, Li et al. [82] proposed a deterministic approach to multiple-source broadcasting. This scheme assumes that each node has local network topology information within its 2-hop neighborhood. Nodes to forward packets encode them according to local network topology information and forward the encoded packets to their neighboring nodes.

Flooding is inherently a highly robust mechanism in terms of packet losses because each receiver node has many opportunities to receive broadcast packets. Therefore, when network coding is applied to broadcasting, the number of forwarded packets is reduced without losing opportunities to retrieve native packets even if some packets are lost. As a result, the main contribution of network coding to the broadcast storm problem is that it can reduce the number of forwarded packets without degradation of packet loss probabilities. Related work on network coding-based broadcasting can be found in Hou et al. [53], Yang and Wu [136], Yang et al. [137].

4. Robustness Enhancement

FEC is an important technique to achieve reliable communications, and it has two roles: *error correction* and *erasure correction*. Usually, FEC is considered as a source-based coding technique, where source nodes append the redundancy to native packets to be sent (e.g., parity packets in linear block codes). As we will see, network coding can play a role similar to FEC, which leads to the robustness enhancement in data transfer.

4.1 Network Error Correcting Codes

Consider a network with a pair of source and receiver nodes. Let $\boldsymbol{x} = (x_1\ x_2\ \cdots\ x_N)$ and $\boldsymbol{y} = (y_1\ y_2\ \cdots\ y_N)$ denote native packets generated at the source node and coded packets received at the receiver node, respectively. Note that $\boldsymbol{y}$ can be expressed to be $\boldsymbol{y} = \boldsymbol{C}\boldsymbol{x} + \boldsymbol{e}$, where $\boldsymbol{C}$ and $\boldsymbol{e}$ denote a coding matrix and the contribution of packet errors to the received packet $\boldsymbol{y}$, respectively. The purpose of network error correcting codes is to recover $\boldsymbol{x}$ from $\boldsymbol{y}$, where spatial redundancies are appended by network coding.

The concept of error correction codes with network coding, referred to as *network error correction codes*, is studied in Cai and Yeung [9], [10], Yeung and Cai [140]. These papers generalize some theoretical bounds in traditional coding theory to network error correcting codes. Zhang [148] defines the minimum distance of a network error correcting code and shows that it plays the same role as it does in traditional coding theory. Construction algorithms of network error correcting codes are studied in Matsumoto [96], Yang et al. [135].

In the above literature, *coherent network coding* is assumed; that is, source and receiver nodes have knowledge about network topologies and coding operations at intermediate nodes. On the other hand, *noncoherent network coding* is considered in Koetter and Kschischang [71], Silva et al. [115], where random network coding is used and source and receiver nodes do not have any knowledge about network topologies and coding operations. These papers consider *subspace codes* that utilize subspace properties of random network coding: any linear combination of vectors in a linear subspace belongs to the same subspace.

In wireless networks, network error correcting codes have other interesting research topics, such as joint network-channel coding (Guo et al. [44], Hausl et al. [47], Hausl and Dupraz [48]) and matching code-on-graph with network-on-graph (Bao and Li [5]). Because these topics contain physical layer issues, we do not discuss them further.

4.2 Erasure Correction

4.2.1 Network Erasure Correcting Codes

Network erasure correcting codes utilize spatial redundancy of network coding for recovering lost packets. Koetter and Médard [70] formulate a robust linear network coding for link failures. Figure 6 shows an example of linear network coding for network erasure correcting codes, where coefficients (c_1, c_2), (c_3, c_4), (c_5, c_6), (c_7, c_8), and (c_9, c_{10}) are assigned to output links of nodes D, E, H, I, and J, respectively. Let $\boldsymbol{x} = (x_1\ x_2\ x_3)^{\mathrm{T}}$ and $\boldsymbol{y} = (y_1\ y_2\ y_3)^{\mathrm{T}}$ denote packets transmitted from source node S and received at receiver node R, respectively. If no packets are lost, we have $\boldsymbol{y} = \boldsymbol{C}\boldsymbol{x}$, where

$$\boldsymbol{C} = \begin{pmatrix} c_5 + c_1c_6 & c_2c_6 & 0 \\ c_1c_7 & c_2c_7 + c_3c_8 & c_4c_8 \\ 0 & c_3c_9 & c_4c_9 + c_{10} \end{pmatrix}.$$

Suppose a packet is lost on link (F,I). $\boldsymbol{C}$ then becomes

$$\boldsymbol{C} = \begin{pmatrix} c_5 + c_1c_6 & c_2c_6 & 0 \\ 0 & c_3c_8 & c_4c_8 \\ 0 & c_3c_9 & c_4c_9 + c_{10} \end{pmatrix}.$$

Therefore $\boldsymbol{x}$ can be retrieved if $\operatorname{rank} \boldsymbol{C} = 3$. If packets are lost both in link (A,H) and in link (F,H), $\boldsymbol{C}$ becomes

$$\boldsymbol{C} = \begin{pmatrix} 0 & 0 & 0 \\ c_1c_7 & c_2c_7 + c_3c_8 & c_4c_8 \\ 0 & c_3c_9 & c_4c_9 + c_{10} \end{pmatrix}.$$

so that $\boldsymbol{x}$ cannot be retrieved because $\operatorname{rank} \boldsymbol{C} < 3$.

Robustness of linear network coding can be improved by combining it with source-based coding (Fragouli and Soljanin [36]). Consider Fig. 6 again, where we assume that source node S transfers two native packets $\boldsymbol{a} = (a_1, a_2)^{\mathrm{T}}$ to receiver node R. Before transmission, source node S generates three coded packets by

$$\boldsymbol{x} = \boldsymbol{G}\boldsymbol{a} = \begin{pmatrix} g_{1,1} & g_{1,2} \\ g_{2,1} & g_{2,2} \\ g_{3,1} & g_{3,2} \end{pmatrix} \begin{pmatrix} a_1 \\ a_2 \end{pmatrix}.$$

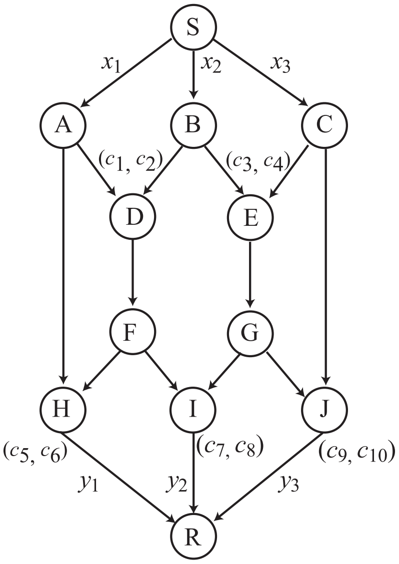


Fig. 6 An example of linear network coding for network erasure correcting codes.

In this case, the native packets $\boldsymbol{a} = (a_1, a_2)^{\mathrm{T}}$ can be retrieved if $\operatorname{rank} \boldsymbol{C}\boldsymbol{G} = 2$ because $\boldsymbol{y} = \boldsymbol{C}\boldsymbol{x} = \boldsymbol{C}\boldsymbol{G}\boldsymbol{a}$. Note that in this specific example, the combination of network coding with source-based coding diminishes the maximum throughput to be 2/3, compared with the system only with network coding. Matsuda and Takine [95] proposed Reed Solomon/network joint coding, where $\boldsymbol{G}$ is set to be a generator matrix of Reed-Solomon erasure correcting codes. Network erasure correcting codes are applied to various network environments of computer networks, e.g., protection in optical networks (Kamal [61], Manley et al. [91]), and wireless sensor and wireless mesh networks (Al-Kofahi and Kamal [4]).

4.2.2 Network Coding-Based Routing

In wireless networks, routing schemes with network erasure correcting codes are studied in order to improve the robustness to packet losses. Kamra et al. [62] proposed *Growth Codes* in order to increase the persistence of sensor networks, which is defined as a fraction of sensed information to be delivered eventually to sink nodes. In Growth Codes, each sensor node determines degree d according to a certain distribution before forwarding a packet. The node then randomly chooses d packets from among its received packets and forwards the coded packet generated by XORing these d packets. Initially, the degree is set to be one, and as time goes by, the degree distribution is updated in such a way that the mean degree increases.

Network coding has also been applied to DTN (Delay/ Disruption/ Disconnection Tolerant Network) environments (Farrell and Cahill [28]), such as sparsely populated mobile ad hoc networks. In these networks, mobile nodes are chronically isolated each other and end-to-end connections can hardly be established at any moment. Therefore the *store-carry-forward* routing paradigm is used to deliver packets, i.e., a node receiving a packet stores it in its buffer, carries it while moving, and forwards it to others when the node encounters them.

In store-carry-forward routing schemes, there is trade-off between data delivery delay and the number of forwarded packets in the network. In Epidemic routing (Vahdat and Becker [123]), a node carrying packets forwards them to every node that it encounters. Although Epidemic routing achieves the shortest data delivery delay, it causes excessive network resource consumption (Matsuda and Takine [93]). In Lin et al. [84], [85], Zhang et al. [146], network coding is combined with store-carry-forward routing schemes in order to decrease the number of forwarded packets. For example, assume that node A has N packets $\boldsymbol{y} = (y_1\ y_2\ \cdots\ y_N)$. When node A encounters another node B, node A encodes packet $\boldsymbol{y}$ into a coded packet $z = \sum_{i=1}^{N} c_i y_i$ and forward z to node B. Related work can be found in Ahmed and Kanhere [2], Subramanian and Fekri [117], Widmer and Le Boudec [126], Zhang et al. [144].

5. Network Tomography

Inferring internal network characteristics using end-to-end measurements are called *network tomography*. So far, many network tomography schemes have been proposed for link loss rates (Cáceres et al. [7], Tsuru et al. [121]), link delays (Lo Presti et al. [88], Tsang et al. [120]), and topology (Coates et al. [20], Ratnasamy and McCanne [104]).

In network coding, coding vectors in coded packets implicitly contain topology information which they passed through. Consider an example in Fig. 7, where nodes A and B transmit packets a and b, respectively, node C encodes these packets into $a + b$, and node D receives the coded packet. When the received packet is equal to $a + b$, node D realizes that there are at least two source nodes and one intermediate node with two input links. Siavoshani et al. [113] show that subspace spanned by coded packets received at each node gives topology information of the network.

In what follows, we describe two kinds of network tomography schemes with network coding: link loss rate inference and topology inference. All schemes utilize the above-mentioned fact that coding vectors in coded packets received at a node indicate which links those packets passed through successfully. In the link loss rate inference, link loss rates are estimated by collecting coding vectors at receiver nodes. On the other hand, in the topology inference, network topology is estimated by collecting coding vectors transmitted among different sets of source and receiver nodes.

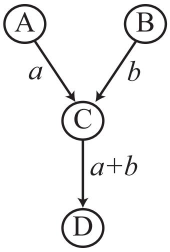


Fig. 7 A toy example for network coding.

5.1 Link Loss Rate Inference

The study of link loss rate inference with network coding was started by Ho et al. [49]. In what follows, we refer to link loss rate inference as *loss tomography*. With the same example as in Fragouli and Markopoulou [31], we explain how network coding can be used for loss tomography. Figure 8 shows a directed tree topology, where nodes A and B transmit packets a and b, respectively. Node C encodes them into packet $a + b$ and forwards it to node D. Nodes E and F receive packets forwarded from node D. We assume that nodes A and B are synchronized and node C receives packets a and b at the same time. As a result, if node C forwards packet a (resp. b) to node D, it implies that packet b (resp. a) is lost on link (B,C) (resp. (A,C)).

Assume that packets are lost on each link independently and randomly. Table 3 shows the relationship between received packets and successful/failed packet transmission on each link. The two leftmost columns denote packets received by nodes E and F, and the five rightmost columns show the corresponding success and failure events on respective links. From the table, the probability of each possible event can be written as a function of loss probabilities of respective links. For example, suppose that node E receives packet a and node F does not receive any packet. This event occurs only when transmitted packets on links (B,C) and (D,F) are lost and the probability of this event is given by $(1-\alpha_{(\mathrm{A,C})})\alpha_{(\mathrm{B,C})}(1-\alpha_{(\mathrm{C,D})})(1-\alpha_{(\mathrm{D,E})})\alpha_{(\mathrm{D,F})}$, where $\alpha_{(\mathrm{X,Y})}$ denotes the packet loss probability of link (X,Y). In this way, we can collect the frequencies of patterns of received packets by injecting packets into the network successively. Link loss rates $\alpha_{(X,Y)}$ would be inferred by means of the maximum likelihood estimation.

The difficulty of loss tomography with network coding depends on network topology. Fragouli and Markopoulou [31] evaluate the performance of loss tomography for the tree topology in Fig. 8. Further Fragouli et al. [34] proposed three algorithms for general tree topologies. Note that tree

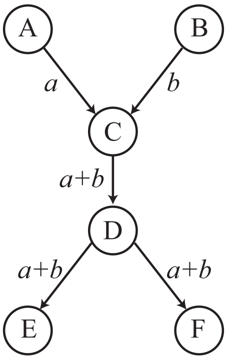


Fig. 8 Link loss rate inference for tree topology (Fragouli and Markopoulou [31]).

Table 3 Relationship between received packets and link states for the tree topology in Fig. 8, where $\emptyset$ denotes the event that no packet is received, and S and F denote a successful transmission and a failed transmission, respectively.

received packets		link states				
node E	node F	(A,C)	(B,C)	(C,D)	(D,E)	(D,F)
$a+b$	$a+b$	S	S	S	S	S
$\emptyset$	$a+b$	S	S	S	F	S
a	a	S	F	S	S	S
$\emptyset$	a	S	F	S	F	S
b	b	F	S	S	S	S
$\emptyset$	b	F	S	S	F	S
$a+b$	$\emptyset$	S	S	S	S	F
a	$\emptyset$	S	F	S	S	F
b	$\emptyset$	F	S	S	S	F
$\emptyset$	$\emptyset$	F	F	–	–	–
		S	S	F	–	–
		S	F	F	–	–
		F	S	F	–	–
		S	S	S	F	F
		S	F	S	F	F
		F	S	S	F	F

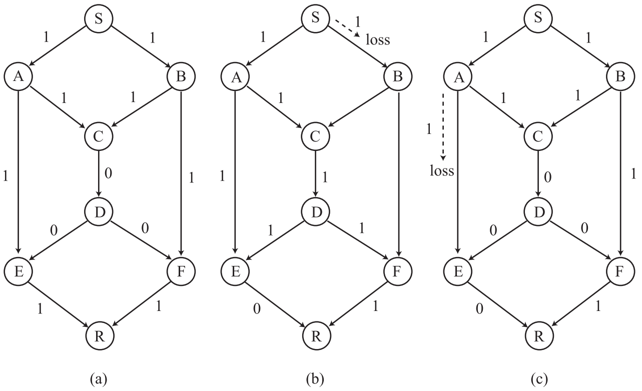
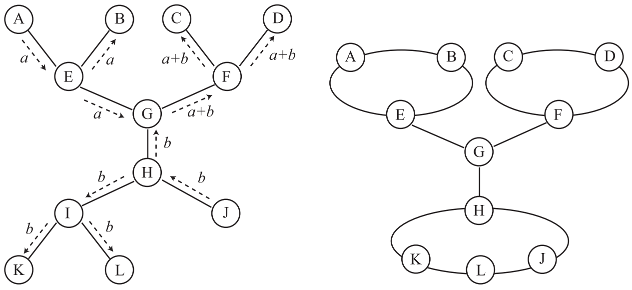


Fig. 9 An example of loss tomography with network coding in non-tree topology, where network coding is defined on **GF**(2). (a) There is no packet loss. (b) A packet is lost on link (S,B). (c) A packet is lost on link (A,E).



(a) Leaf nodes A and J are source nodes and the remaining leaf nodes are receiver nodes.

(b) Estimated topology after one iteration.

Fig. 10 An example of topology inference with network coding (Fragouli et al. [33]).

topology is the simplest case, where routes from respective output links of a source node to all receiver nodes form directed trees.

Gjoka et al. [38] proposed a loss tomography scheme with network coding for general non-tree topology. Link loss rates are estimated with the *sum-product algorithm* (Kschischang et al. [73]) according to the same formulation as in Mao et al. [92]. Ho et al. [49] proposed a passive loss tomography with random network coding, where the probability of distinguishing failure patterns is also derived. Lin et al. [86] proposed a passive loss tomography scheme with network coding for wireless sensor networks. See Gui et al. [43] also.

Similar to other applications, we have to select the size q of finite field $\mathbf{GF}(q)$ used in loss tomography. In the case of tree topology, $\mathbf{GF}(2)$ (i.e., XOR operations) is enough to estimate link loss rates. In general, however, we have to use $\mathbf{GF}(q)$ with q large enough (Fragouli and Markopoulou [31], Fragouli et al. [34], Gjoka et al. [38]). Figure 9 shows an example for loss tomography in non-tree topology, where $\mathbf{GF}(2)$ is employed. When only node R is a receiver node, single link losses in links (S,B) and (A,E) cannot be distinguished, as shown in Fig. 9(b) and (c). Ho et al. [49] discuss the minimum required size of $\mathbf{GF}(q)$ in loss tomography with random network coding. Also, necessary and sufficient conditions for identifiability is given in Fragouli and Markopoulou [31, Theorem 1], where link (v,u) is *identifiable* if it is possible to estimate the associated link loss rate $\alpha_{(v,u)}$ when the size of finite field is large enough.

5.2 Topology Inference

Fragouli et al. [33] describe the following basic idea of topology inference with network coding. Consider a tree topology in Fig. 10, which is the same example as in Fragouli et al. [33]. We estimate network topology iteratively by dividing nodes in the network into groups. Note that in each iteration, a different set of leaf nodes serves as source nodes. For example, in the first iteration, leaf nodes A and J are selected as source nodes and the remaining leaf nodes as receiver nodes. Nodes A and J send packets a and b, respectively. Node G encodes packets a and b into packet $a+b$ and forwards it to node F. Other intermediate nodes duplicate received packets and forward them to their children. Therefore, node B receives packet a, nodes C and D receive packet $a+b$, and nodes J, K, and L receive packet b. Leaf nodes are then divided into three groups according to their received packets: $\mathcal{L}_1 = \{A, B\}$, $\mathcal{L}_2 = \{C, D\}$, $\mathcal{L}_3 = \{J, K, L\}$. As a result, we can deduce that the network topology has the structure shown in Fig. 10(b). In each of the subsequent iterations, a different set of source nodes are chosen and the above procedure is applied repeatedly.

Sattari et al. [107] proposed a topology inference scheme for general topologies with multiple source and multiple receiver nodes. Yao et al. [138] discuss topology inference with link failure.

6. Security

In this section, we discuss two security issues, pollution attacks and eavesdropping, in network systems with network coding. Those issues are inherent in network coding and in general, they are orthogonal to other applications presented in this survey. A comprehensive discussion on security issues in network coding can be found in [25].

6.1 Pollution Attacks

As mentioned in Sect. 1.3.2, network coding is vulnerable to pollution attacks. This can cause a serious problem, especially in multicast communications, because a few corrupted packets can yield a large number of corrupted coded packets disseminated over receiver nodes.

With the symbol-based format introduced in Sect. 1.2, pollution attacks can be modeled as follows (Fragouli and Soljanin [36], Jaggi et al. [59]). Consider a network with a pair of a source and receiver nodes. Let L denote the number of symbols in a packet. We define an $N \times L$ matrix $\boldsymbol{X}$, an $N \times L$ matrix $\boldsymbol{Y}$, and an $N_\mathrm{p} \times L$ matrix $\boldsymbol{Z}$ as native packets, coded packets received at the receiver node, and packets injected by malicious nodes, respectively.

$$\boldsymbol{X} = \begin{pmatrix} \boldsymbol{x}_1 \\ \boldsymbol{x}_2 \\ \vdots \\ \boldsymbol{x}_N \end{pmatrix}, \quad \boldsymbol{Y} = \begin{pmatrix} \boldsymbol{y}_1 \\ \boldsymbol{y}_2 \\ \vdots \\ \boldsymbol{y}_N \end{pmatrix}, \quad \boldsymbol{Z} = \begin{pmatrix} \boldsymbol{z}_1 \\ \boldsymbol{z}_2 \\ \vdots \\ \boldsymbol{z}_{N_\mathrm{p}} \end{pmatrix}.$$

Because $\boldsymbol{Z}$ is mixed up linearly with $\boldsymbol{X}$ on the way to the receiver node, $\boldsymbol{Y}$ can be expressed to be

$$\boldsymbol{Y} = \boldsymbol{C}\boldsymbol{X} + \boldsymbol{C}'\boldsymbol{Z}, \tag{4}$$

where $\boldsymbol{C}$ denotes an $N \times N$ coding matrix for $\boldsymbol{x}$, which corresponds to paths from the source node to the receiver node, and $\boldsymbol{C}'$ denotes an $N \times N_\mathrm{p}$ coding matrix for $\boldsymbol{z}$, which corresponds to paths from malicious nodes to the receiver node.

There are two approaches to preventing pollution attacks: *correcting corrupted packets* and *detecting corrupted packets*. The former approach aims to correct corrupted packets at receiver nodes, which will be explained in Sect. 6.1.1. On the other hand, the latter approach aims to detect and discard corrupted packets at intermediate and receiver nodes, which will be explained in Sect. 6.1.2.

6.1.1 Correcting Corrupted Packets

As discussed in Sect. 4.1, corrupted packets can be corrected with network error correcting codes. However, Jaggi et al. [59] proposed a different scheme that a source node sends parity check information to receiver nodes. See Fragouli and Soljanin [36] also. In what follows, we explain a shared secret model in Jaggi et al. [59].

Suppose the source node intends to transmit N original packets $\boldsymbol{x}_{\mathrm{org},i}$ ($i = 1, 2, \ldots, N$), where all $\boldsymbol{x}_{\mathrm{org},i}$ ($i = 1, 2, \ldots, N$) are composed of L_org symbols in a sufficiently large $\mathbf{GF}(q)$. The source node first constructs N native packets $\boldsymbol{X}$ by appending unit matrix. i.e.,

$$\boldsymbol{X} = (\boldsymbol{X}_\mathrm{org}\ \boldsymbol{I}), \tag{5}$$

where $\boldsymbol{X}_\mathrm{org}$ denotes an $N \times L_\mathrm{org}$ matrix whose ith row is given by $\boldsymbol{x}_{\mathrm{org},i}$, and $\boldsymbol{I}$ denotes an $N \times N$ unit matrix. Thus each of the native packets contains $L = L_\mathrm{org} + N$ symbols in total and all of them are sent to receiver nodes.

Also, the source node randomly choose N symbols g_i ($i = 1, 2, \ldots, N$) from $\mathbf{GF}(q)$ and prepares an $L_\mathrm{org} \times \eta$ parity check matrix $\boldsymbol{H}$ whose (i, j)th ($i = 1, 2, \ldots, L_\mathrm{org}$, $j = 1, 2, \ldots, \eta$) element is given by g_i^{j-1}, where η is a design parameter. With $\boldsymbol{H}$, the source node produces hashed information $\boldsymbol{P} = \boldsymbol{X}_\mathrm{org}\boldsymbol{H}$, and using a secret channel, the source node sends g_i ($i = 1, 2, \ldots, N$) and $\boldsymbol{P}$ to receiver nodes.

Each receiver node corrects corrupted packets $\boldsymbol{Y}$ in Eq. (4), using $\boldsymbol{H}$ and $\boldsymbol{P}$ in the following way. Note here that when coding information is embedded in the packet header of coded packets, the receiver node cannot know $\boldsymbol{C}$ because the packet header is also corrupted. It follows from Eqs. (4) and (5) that $\boldsymbol{Y} = (\boldsymbol{Y}_1\ \boldsymbol{Y}_2)$ is given in terms of $\boldsymbol{Z} = (\boldsymbol{Z}_1\ \boldsymbol{Z}_2)$:

$$\boldsymbol{Y}_1 = \boldsymbol{C}\boldsymbol{X}_\mathrm{org} + \boldsymbol{C}'\boldsymbol{Z}_1, \quad \boldsymbol{Y}_2 = \boldsymbol{C} + \boldsymbol{C}'\boldsymbol{Z}_2,$$

where $\boldsymbol{Y}_1$ and $\boldsymbol{Z}_1$ denote $N \times L_\mathrm{org}$ matrices and $\boldsymbol{Y}_2$ and $\boldsymbol{Z}_2$ denote $N \times N$ matrices. Because $\boldsymbol{C} = \boldsymbol{Y}_2 - \boldsymbol{C}'\boldsymbol{Z}_2$, $\boldsymbol{Y}_1$ is rewritten to be

$$\boldsymbol{Y}_1 = \boldsymbol{Y}_2\boldsymbol{X}_\mathrm{org} + \boldsymbol{E}_1, \tag{6}$$

where $\boldsymbol{E}_1 = \boldsymbol{C}'(\boldsymbol{Z}_1 - \boldsymbol{Z}_2\boldsymbol{X})$ denotes an unknown $N \times L_\mathrm{org}$ matrix.

The unknown $\boldsymbol{E}_1$ in Eq. (6) is obtained as follows. Post-multiplying both sides of Eq. (6) by $\boldsymbol{H}$, using $\boldsymbol{X}_\mathrm{org}\boldsymbol{H} = \boldsymbol{P}$, and rearranging terms yield

$$\boldsymbol{E}_1\boldsymbol{H} = \boldsymbol{Y}_1\boldsymbol{H} - \boldsymbol{Y}_2\boldsymbol{P}. \tag{7}$$

It is shown in Jaggi et al. [59, Claim 5] that for any $\boldsymbol{E}'_1 \neq \boldsymbol{E}$, the probability of $\boldsymbol{E}'_1\boldsymbol{H} = \boldsymbol{E}_1\boldsymbol{H}$ is not greater than $(N/q)^\eta$. Therefore, if the size q of finite field $\mathbf{GF}(q)$ is enough large, Eq. (7) has a unique solution for $\boldsymbol{E}_1$ with high probability. Once $\boldsymbol{E}_1$ is obtained, the receiver node can retrieve $\boldsymbol{X}_\mathrm{org}$ by solving Eq. (6).

6.1.2 Detection of Pollution Attacks

In this approach, intermediate nodes and receiver nodes detect pollution attacks from received packets and discard corrupted packets if pollution attacks are detected. Existing schemes for detecting pollution attacks are categorized into *cryptographic schemes* and *subspace properties-based schemes*, which will be discussed below.

(1) Cryptographic Schemes

Ho et al. [51] proposed a cryptographic scheme with polynomial hash functions in order to detect pollution attacks at receiver nodes. A polynomial hash function is a polynomial function to generate a hash symbol from data symbols in a packet. Suppose a source node transmits N native packets to receiver nodes. For each of those packets, the source node first computes redundant hash symbols by a specific polynomial hash function of data symbols in the native packet and appends them to the original native packet. After that, the source node constructs N coded packets by random network coding with N native packets and transmits them to receiver nodes. Each receiver node decodes a set of received coded packets and then obtains hash values by the polynomial hash function of the resulting data symbols. If any of them are different from the corresponding hash values in the retrieved packets, the receiver node considers that one or more packets are corrupted.

Cryptographic schemes with homomorphic hash function (or homomorphic signature functions) were also proposed in Gkantsidis and Rodriguez [40], Yu et al. [142]. Homomorphic hash functions are suitable for linear network coding because the hash value $h(\boldsymbol{p})$ of a linearly coded output packet $\boldsymbol{p} = \sum_{i=1}^{N} c_i \boldsymbol{x}_i$ is given in terms of those of input packets $\boldsymbol{x}_i$, i.e.,

$$h(\boldsymbol{p}) = \prod_{i=1}^{N} h^{c_i}(\boldsymbol{x}_i). \tag{8}$$

For example, the scheme in Yu et al. [142] assumes that the source node calculates a signature $h(\boldsymbol{x}_i)$ $(i = 1, 2, \ldots, N)$ for native packet $\boldsymbol{x}_i$ with its private key and appends it to the original packet $\boldsymbol{x}_i$. When intermediate nodes and receiver nodes receive a packet, they check if the data symbols are consistent with their signature, using the public key. When intermediate nodes receive a set of consistent packets, those packets are linearly encoded, along with the signature calculated by Eq. (8), and forwarded to neighboring nodes. See Charles et al. [14], Gkantsidis and Rodriguez [40], Zhao et al. [149] also.

(2) Subspace Properties-Based Schemes

Kehdi and Li [65] proposed a detection scheme for pollution attacks, using subspace properties. Let $\boldsymbol{X}$ denote a matrix of native packets, as defined in Eq. (1). Native packets $\boldsymbol{x}_i$ $(i = 1, 2, \ldots, N)$ are composed of $L + N$ symbols, where the first L symbols correspond to information to be delivered to the receiver node and the remaining N symbols correspond to augmented symbols in order to detect pollution attacks.

Let $\Pi_{\boldsymbol{X}}$ denote a subspace spanned by $\boldsymbol{X}$ and let $\Pi_{\boldsymbol{X}}^{\perp}$ denote the null space spanned by a set of all vectors z such that $\boldsymbol{X}z = 0$. This scheme uses subspace properties of linear network coding, where any linear combination of vectors in a linear subspace belongs to the same subspace, and all vectors in $\Pi_{\boldsymbol{X}}$ are orthogonal to any linear combination of vectors in $\Pi_{\boldsymbol{X}}^{\perp}$. Intermediate and receiver nodes are provided with a set of vectors in $\Pi_{\boldsymbol{X}}^{\perp}$, and detect pollution attacks by using the orthogonality. Subspace properties are also used to locate links at which pollution attacks occur (Siavoshani et al. [114]).

6.2 Eavesdropping

Suppose a source node transmits N packets to a receiver node without network coding and an adversary eavesdrops k packets among those. In this case, the source node can communicate with the receiver node securely, using a secure wiretap channel code. For example, if an $(N, N - k)$ linear MDS code is employed, the source node can send $N - k$ packets to the receiver node securely. In network coding, however, linear operations of packets may undermine security even if the secure wiretap channel code is employed. See Fragouli and Soljanin [36, Example 7.4].

In order to solve this problem, Cai et al. [8] define a communication system on a wiretap network, which is a model incorporating network coding into information security, and show that for a given k $(k < N)$, there exists an appropriate code transformation that guarantees the secure delivery of $N - k$ packets to the receiver, which is called k-secure network coding. Harada and Yamamoto [46] consider a stronger version of k-secure network coding. Feldman et al. [30] show that k-secure network coding can be achieved if input packets are encoded appropriately.

Formally, the problem of eavesdropping is formulated as follows (Fragouli and Soljanin [36]). Let $\boldsymbol{x} = (x_1\ x_2\ \cdots\ x_N)$ denote original packets to be delivered to receiver nodes, and let $\boldsymbol{y} = (y_1\ y_2\ \cdots\ y_N)$ and $z = (z_1\ z_2\ \cdots\ z_{N_e})$ denote coded packets received at a receiver node and at an eavesdropper, respectively. A network coding scheme is called *information theoretically secure* if the eavesdropper cannot obtain any information about $\boldsymbol{x}$ from z in a sense that

$$\mathcal{H}(\boldsymbol{x} \mid \boldsymbol{y}) = 0 \quad \text{and} \quad \mathcal{I}(\boldsymbol{x}, z) = 0,$$

where $\mathcal{H}(\boldsymbol{x} \mid \boldsymbol{y})$ and $\mathcal{I}(\boldsymbol{x}, z)$ denote the conditional entropy of $\boldsymbol{x}$ given $\boldsymbol{y}$ and the mutual information between $\boldsymbol{x}$ and z, respectively.

On the other hand, a secure network coding scheme is called *weakly secure* if the eavesdropper cannot obtain any information about x_i from z.

$$\mathcal{H}(\boldsymbol{x} \mid \boldsymbol{y}) = 0 \quad \text{and} \quad \mathcal{I}(x_i, z) = 0 \quad (i = 1, 2, \ldots, N).$$

In weakly secure systems, the eavesdropper cannot obtain meaningful information about individual native packets from coded packets available to him. Weakly secure network coding schemes are studied in Bhattad and Narayanan [6], Silva et al. [116], which can deliver N packets to receivers in a sense of weakly security.

7. Concluding Remarks

In this survey, we presented an overview of the research on network coding. We first classified the network coding research into network coding design problems and network coding applications. Furthermore, we divided network coding applications into four research areas, throughput/capacity enhancement, robustness enhancement, network tomography, and security. Network coding was proposed originally in order to achieve multicast data transfer at the max-flow rate and it has mainly been studied as throughput/capacity enhancement techniques, especially in wireless and overlay networks.

On the other hand, network error/erasure correcting codes effectively utilize spatial redundancy of network coding so as to improve robustness, instead of improving the throughput performance. Network tomography utilizes the relationship between network topology and coding matrices in order to infer the internal characteristics of networks, which might expand network monitoring capabilities. Security issues in network coding will become more important when network coding is deployed widely. Although we did not discuss network coding in the physical layer, it may also include interesting research topics, such as MIMO network coding (Fasolo et al. [29], Tran et al. [119], Xu et al. [132]) and cooperative communication with network coding (Chen et al. [16], Xiao et al. [131]).

The many research areas in network coding suggest the enormous potential of network coding. In the conventional architecture of IP networking, however, intermediate nodes should be dedicated to simple operations in the network layer, i.e., routing and forwarding. Therefore, in order to diffuse network coding to many areas in computer networking, we have to think out *why intermediate nodes should encode packets*. The authors of this survey are of the opinion that network coding should be applied to applications for which source-based coding is not suitable. As introduced in this survey, there ought to be many such applications that network coding is one of the best approaches to enhance their performance.

Acknowledgement

Research of the first author was supported in part by Grand-in-Aid for Young Scientists (B) of the Ministry of Education, Culture, Sports, Science and Technology (MEXT) under Grant No. 22700069 and the Telecommunications Advancement Foundation (TAF). Research of the second author was supported in part by Grand-in-Aid for Young Scientists (B) of the Ministry of Education, Culture, Sports, Science and Technology (MEXT) under Grant No. 21700089 and the Nakajima Foundation. Research of the third author was supported in part by Grant-in-Aid for Scientific Research (C) of Japan Society for the Promotion of Science under Grant No. 22500056.

References

[1] R. Ahlswede, N. Cai, S.-Y.R. Li, and R.W. Yeung, “Network information flow,” IEEE Trans. Inf. Theory, vol.46, no.4, pp.1204–1206, July 2000.

[2] S. Ahmed and S.S. Kanhere, “HUBCODE: Message forwarding using hub-based network coding in delay tolerant networks,” Proc. 12th ACM Int’l Conf. Modeling, Anal. and Simu. of Wireless and Mobi. Sys. (MSWiM 2009), pp.288–296, Canary Islands, Spain, Oct. 2009.

[3] R.K. Ahuja, T.L. Magnanti, and J.B. Orlin, Network Flows, Theory, Algorithms, and Applications, Prentice Hall, Englewood Cliffs, NJ, 1993.

[4] O.M. Al-Kofahi and A.E. Kamal, “Network coding-based protection of many-to-one wireless flows,” IEEE J. Sel. Areas Commun., vol.27, no.5, pp.797–813, June 2009.

[5] X. Bao and J. Li, “Adaptive network coded cooperation (ANCC) for wireless relay networks: matching code-on-graph with network-on-graph,” IEEE Trans. Wirel. Commun., vol.7, no.2, pp.574–583, Feb. 2008.

[6] K. Bhattad and K.R. Narayanan, “Weakly secure network coding,” Proc. 1st Workshop on Netw. Coding, Theory and Applicat. (NETCOD 2005), Riva del Garda, Italy, April 2005.

[7] R. Cáceres, N.G. Duffield, J. Horowitz, and D.F. Towsley, “Multicast-based inference of network-internal loss characteristics,” IEEE Trans. Inf. Theory, vol.45, no.7, pp.2462–2480, Nov. 1999.

[8] N. Cai and R.W. Yeung, “Secure network coding,” Proc. 2002 IEEE Int’l Symp. on Inf. Theory (ISIT 2002), p.323, Lausanne, Switzerland, June/July 2002.

[9] N. Cai and R.W. Yeung, “Network coding and error correction,” Proc. 2002 IEEE Inf. Theory Workshop, pp.119–122, Dublin, Ireland, Oct. 2002.

[10] N. Cai and R.W. Yeung, “Network error correction, part II: lower bounds,” Commun. Inf. and Sys., vol.6, no.1, pp.37–54, 2006.

[11] M. Celebiler and G. Stette, “On increasing the down-link capacity of a regenerative satellite repeater in point-to-point communications,” Porc. IEEE, vol.66, no.1, pp.98–100, Jan. 1978.

[12] S. Chachulski, M. Jennings, S. Katti, and D. Katabi, “Trading structure for randomness in wireless opportunistic routing,” Proc. ACM SIGCOMM 2007, pp.169–180, Kyoto, Japan, Aug. 2007.

[13] P. Chaporkar and A. Proutiere, “Adaptive network coding and scheduling for maximizing throughput in wireless networks,” Proc. ACM MOBICOM 2007, pp.135–146, Montréal, Canada, Sept. 2007.

[14] D. Charles, K. Jain, and K. Lauter, “Signatures for network coding,” Proc. 40th Annu. Conf. Inf. Sci. and Sys., pp.857–863, Princeton, NJ, March 2006.

[15] C. Chekuri, C. Fragouli, and E. Soljanin, “On average throughput and alphabet size in network coding,” IEEE Trans. Inf. Theory, vol.52, no.6, pp.2410–2424, June 2006.

[16] Y. Chen, S. Kishore, and J. Li, “Wireless diversity through network coding,” Proc. IEEE WCNC 2006, vol.3, pp.1681–1686, Las Vegas, NV, April 2006.

[17] X. Chenguang, X. Yinlong, Z. Cheng, W. Ruizhe, and W. Qingshan, “On network coding based multirate video streaming in directed networks,” Proc. IEEE Int’l Conf. Perfor. Comput. and Commun. (IPCCC’07), pp.332–339, New Orleans, LA, April 2007.

[18] D.M. Chiu, R.W. Yeung, J. Huang, and B. Fan, “Can network coding help in P2P networks?,” Proc. 4th Int’l Symp. Modelling and Optimization in Mobile, Ad Hoc and Wireless Networks, Boston, MA, April 2006.

[19] P.A. Chou, Y. Wu, and K. Jain, “Practical network coding,” Proc. 41st Allerton Conf. on Commun., Control, and Comput. Monticello, IL, Oct. 2003.

[20] M. Coates, R. Castro, R. Nowak, M. Gadhiok, R. King, and Y. Tsang, “Maximum likelihood network topology identification from edge-based unicast measurements,” ACM SIGMETRICS Perfor. Eval. Review, vol.30, no.1, pp.11–20, June 2002.

[21] B. Cohen, “Incentives build robustness in bittorrent,” Proc. 1st Workshop on Econ. of Peer-to-Peer Sys., Berkeley, CA, June 2003.

[22] Y. Cui and K. Nahrstedt, “Layered peer-to-peer streaming,” Proc. ACM NOSSDAV 2003, pp.162–171, Monterey, CA, June 2003.

[23] A.G. Dimakis, P.B. Godfrey, M.J. Wainwright, and K. Ramchandran, “Network coding for distributed storage system,” Proc. IEEE INFOCOM 2007, pp.2000–2008, Anchorage, Alaska, May 2007.

[24] A.G. Dimakis, K. Ramchandran, Y. Wu, and C. Suh, “A survey

on network codes for distributed storage," arXiv:1004.4438, April 2010, Available: http://arxiv.org/abs/1004.4438

[25] J. Dong, R. Curtmola, and C. Nita-Rotaru, "Secure network coding for wireless mesh networks: Threats, challenges, and directions," Comput. Commun., vol.32, no.17, pp.1790–1801, Nov. 2009.

[26] R. Dougherty, C. Freiling, and K. Zeger, "Insufficiency of linear coding in network information flow," IEEE Trans. Inf. Theory, vol.51, no.8, pp.2745–2759, Aug. 2005.

[27] S. Dumitrescu, M. Shao, and X. Wu, "Layered multicast with interlayer network coding," Proc. IEEE INFOCOM 2009, pp.442–449, Rio de Janeiro, Brazil, April 2009.

[28] S. Farrell and V. Cahill, Delay- and Disruption-Tolerant Networking, Artech House, Boston, MA, 2006.

[29] E. Fasolo, F. Rossetto, and M. Zorzi, "Network coding meets MIMO," Proc. 4th Workshop on Net. Coding, Theory and Applicat. (NETCOD 2008), Hong Kong, China, Jan. 2008.

[30] J. Feldman, T. Malkin, R.A. Servedio, and C. Stein, "On the capacity of secure network coding," Proc. 42nd Allerton Conf. Commun., Control, and Comput., Monticello, IL., Sept./Oct. 2004.

[31] C. Fragouli and A. Markopoulou, "A network coding approach to overlay network monitoring," Proc. 43rd Allerton Conf. Commun., Control, and Comput., pp.774–783, Monticello, IL, Sept. 2005.

[32] C. Fragouli and E. Soljanin, "Information flow decomposition for network coding," IEEE Trans. Inf. Theory, vol.52, no.3, pp.829–848, March 2006.

[33] C. Fragouli, A. Markopoulou, and S. Diggavi, "Topology inference using network coding," Proc. 44th Allerton Conf. Commun., Control, and Comput., Monticello, IL, Sept. 2006.

[34] C. Fragouli, A. Markopoulou, R. Srinivasan, and S. Diggavi, "Network monitoring: it depends on your points of view," Proc. 2007 Inf. Theory and Applicat. Workshop, San Diego, CA, Jan. 2007.

[35] C. Fragouli and E. Soljanin, "Network coding fundamentals," Found. and Trends in Netw., vol.2, no.1, pp.1–133, 2007.

[36] C. Fragouli and E. Soljanin, "Network coding applications," Found. and Trends in Netw., vol.2, no.2, pp.135–269, 2007.

[37] C. Fragouli, J. Widmer, and J.-Y. Le Boudec, "Efficient broadcasting using network coding," IEEE/ACM Trans. Netw., vol.16, no.2, pp.450–463, April 2008.

[38] M. Gjoka, C. Fragouli, P. Sattari, and A. Markopoulou, "Loss tomography in general topologies with network coding," Proc. IEEE GLOBECOM 2007, pp.381–386, Washington, DC, Nov. 2007.

[39] C. Gkantsidis and P.R. Rodriguez, "Network coding for large scale content distribution," Proc. IEEE INFOCOM 2005, vol.4, pp.2235–2245, Miami, FL, March 2005.

[40] C. Gkantsidis and P.R. Rodriguez, "Cooperative security for network coding file distribution," Proc. IEEE INFOCOM 2006, pp.1–13, Barcelona, Spain, April 2006.

[41] C. Gkantsidis, J. Miller, and P.R. Rodriguez, "Anatomy of a P2P content distribution system with network coding," Proc. 5th Int'l Workshop on Peer-to-Peer Sys. (IPTPS 2006), Santa Barbara, CA, Feb. 2006.

[42] C. Gkantsidis, J. Miller, and P.R. Rodriguez, "Comprehensive view of a live network coding p2p system," Proc. Internet Measurement Conf. (IMC 2006), pp.177–187, Rio de Janeiro, Brazil, Oct. 2006.

[43] J. Gui, V. Shah-Mansouri, and V.W.S. Wong, "A linear algebraic approach for loss tomography in mesh topologies using network coding," Proc. IEEE ICC 2010, Cape Town, South Africa, May 2010.

[44] Z. Guo, J. Huang, B. Wang, and J.-H. Cui, S. Zhou, and P. Willett, "A practical joint network-channel coding scheme for reliable communication in wireless networks," Proc. ACM MOBICOM 2009, pp.279–288, Beijing, China, May 2009.

[45] P. Gupta and P.R. Kumar, "The capacity of wireless networks," IEEE Trans. Inf. Theory, vol.46, no.2, pp.388–404, March 2000.

[46] K. Harada and H. Yamamoto, "Strongly secure linear network coding," IEICE Trans. Fundamentals, vol.E91-A, no.10, pp.2720–2728, Oct. 2008.

[47] C. Hausl, F. Schreckenbach, I. Oikonomidis, and G. Bauch, "Iterative network and channel decoding on a tanner graph," Proc. 43rd Allerton Conf. Commun., Control, and Comput., Monticello, IL, Sept. 2005.

[48] C. Hausl and P. Dupraz, "Joint network-channel coding for the multiple-access relay channel," Proc. IEEE SECON 2006, pp.817–822, Reston, VA, Sept. 2006.

[49] T. Ho, B. Leong, Y.-H. Chang, Y. Wen, and R. Koetter, "Network monitoring in multicast networks using network coding," Proc. 2005 IEEE Int'l Symp. Inf. Theory (ISIT 2005), pp.1977–1981, Adelaide, Australia, Sept. 2005.

[50] T. Ho, M. Médard, R. Koetter, D.R. Karger, M. Effros, J. Shi, and B. Leong, "A random linear network coding approach to multicast," IEEE Trans. Inf. Theory, vol.52, no.10, pp.4413–4430, Oct. 2006.

[51] T. Ho, B. Leong, R. Koetter, M. Médard, M. Effros, and D.R. Karger, "Byzantine modification detection in multicast networks with random network coding," IEEE Trans. Inf. Theory, vol.54, no.6, pp.2798–2803, June 2008.

[52] T. Ho and D.S. Lun, Network Coding: An Introduction, Cambridge University Press, Cambridge, UK, 2008.

[53] I.-H. Hou, Y.-E. Tsai, F. Abdelzaher, and I. Gupta, "AdapCode: adaptive network coding for code updates in wireless sensor networks," Proc. IEEE INFOCOM 2008, pp.1517–1525, Phoenix, AZ, April 2008.

[54] IEEE International Symposium on Network Coding (NETCOD), http://www.networkcoding.org/

[55] IEEE Trans. Inf. Theory, vol.52, no.6, June 2006.

[56] IEEE Wireless Network Coding Workshop (WiNC), http://winc2010.org/

[57] K. Iwama, H. Nishimura, M. Paterson, R. Raymond, and S. Yamashita, "Polynomial-time construction of linear network coding," Proc. 35th Int'l Colloquium on Automata, Languages and Programming (ICALP 2008), LNCS 5125, pp.271–282, July 2008.

[58] S. Jaggi, P. Sanders, P.A. Chou, M. Effros, S. Egner, K. Jain, and L.M.G.M. Tolhuizen, "Polynomial time algorithms for multicast network code construction," IEEE Trans. Inf. Theory, vol.51, no.6, pp.1973–1982, June 2006.

[59] 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.

[60] S. Jingjing, Z. Bojin, and C. Anni, "Optimal layered multicast using network coding," Proc. 4th Workshop on Netw. Coding, Theory and Applicat. (NETCOD 2008), Hong Kong, China, Jan. 2008.

[61] A.E. Kamal, "$1 + N$ network protection for mesh networks: network coding-based protection using over p-cycles," IEEE/ACM Trans. Netw., vol.18, no.1, pp.67–80, Feb. 2010.

[62] A. Kamra, V. Misra, J. Feldman, D. Rubenstein, "Growth codes: maximizing sensor network data persistence," Proc. ACM SIGCOMM 2006, pp.255–266, Pisa, Italy, Sept. 2006.

[63] S. Katti, S. Gollakota, and D. Katabi, "Embracing wireless interference: analog network coding," Proc. ACM SIGCOMM 2007, pp.397–408, Kyoto, Japan, Aug. 2007.

[64] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Médard, and J. Crowcroft, "XORs in the air, practical wireless network coding," IEEE Trans. Netw., vol.16, no.3, pp.497–510, June 2008.

[65] E. Kehdi and B. Li, "Null keys: limiting malicious attacks via null space properties of network coding," Proc. IEEE INFOCOM 2009, pp.1224–1232, Rio de Janeiro, Brazil, April 2009.

[66] A. Keshavarz-Haddadt and R. Riedi, "Bounds on the benefit of network coding: Throughput and energy saving in wireless networks," Proc. IEEE INFOCOM 2008, pp.376–384, Phoenix, AZ, April 2008.

[67] M. Kim, M. Médard, U.-M. O'Reilly, and D. Traskov, "An evolutionary approach to inter-session network coding," Proc. IEEE INFOCOM 2009, pp.450–458, Rio de Janeiro, Brazil, April 2009.

[68] Y. Kim and G. de Vaciana, "Is rate adaptation beneficial for intersession network coding?," IEEE J. Sel. Areas Commun., vol.27,

no.5, pp.635–646, June 2009.

[69] M. Kim, D. Lucani, X. Shi, F. Zhao, and M. Médard, "Network coding for multi-resolution multicast," Proc. IEEE INFOCOM 2010, San Diego, CA, March 2010.

[70] R. Koetter and M. Médard, "An algebraic approach to network coding," IEEE/ACM Trans. Netw., vol.11, no.5, pp.782–795, Oct. 2003.

[71] R. Koetter and F.R. Kschischang, "Coding for errors and erasures in random network coding," IEEE Trans. Inf. Theory, vol.54, no.8, pp.3579–3591, Aug. 2008.

[72] O. Kosut, L. Tong, and D. Tse, "Nonlinear network coding is necessary to combat general Byzantine attacks," Proc. 47th Allerton Conf. Commun., Control, and Comput., pp.593–599, Monticello, IL, Oct. 2009.

[73] F.R. Kschischang, B.J. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," IEEE Trans. Inf. Theory, vol.47, no.2, pp.498–519, Feb. 2001.

[74] J.F. Kurose and K.W. Ross, Computer Networking: A Top-Down Approach, 5th ed., Addison Wesley, Boston, MA, 2009.

[75] M. Langberg, A. Sprintson, and J. Bruck, "The encoding complexity of network coding," IEEE Trans. Inf. Theory, vol.52, no.6, pp.2386–2397, June 2006.

[76] M. Langberg, A. Sprintson, and J. Bruck, "Network coding: A computational perspective," IEEE Trans. Inf. Theory, vol.55, no.1, pp.147–157, Jan. 2009.

[77] M. Langberg and A. Sprintson, "Recent results on the algorithmic complexity of network coding," Proc. 5th Workshop on Netw. Coding, Theory, and Applicat. (NetCod 2009), Lausanne, Switzerland, June 2009.

[78] J. Le, J.C.S. Lui, and D.-M. Chiu, "DCAR: Distributed coding-aware routing in wireless networks," IEEE Trans. Mobile Comput., vol.9, no.4, pp.596–608, April 2010.

[79] A.R. Lehman and E. Lehman, "Complexity classification of network information flow problems," Proc. 15th Annu. ACM-SIAM Symp. Discrete Algorithms (SODA'04), pp.142–150, New Orleans, LA, 2004.

[80] S.-Y.R. Li, R.W. Yeung, and N. Cai, "Linear network coding," IEEE Trans. Inf. Theory, vol.29, no.2, pp.371–381, Feb. 2003.

[81] Z. Li and B. Li, "Network coding in undirected networks," Proc. 38th Annu. Conf. Inf. Sci. and Sys. (CISS 2004), pp.257–262, March 2004.

[82] L. Li, R. Ramjee, M. Buddhikot, and S. Miller, "Network coding-based broadcast in mobile ad hoc networks," Proc. IEEE INFOCOM 2007, pp.1739–1747, Anchorage, AL, May 2007.

[83] Q. Li, S.H. Ting, and C.K. Ho, "Nonlinear network code for high throughput broadcasting with retransmissions," Proc. 2009 IEEE Int'l Symp. Inf. Theory (ISIT 2009), pp.2853–2857, Seoul, Korea, June 2009.

[84] Y. Lin, B. Li, and B. Liang, "Efficient network coded data transmissions in disruption tolerant networks," Proc. IEEE INFOCOM 2008, pp.2180–2188, Phoenix, AZ, April 2008.

[85] Y. Lin, B. Li, and B. Liang, "Stochastic analysis of network coding in epidemic routing," IEEE J. Sel. Areas Commun., vol.26, no.5, pp.794–808, June 2008.

[86] Y. Lin, B. Liang, and B. Li, "Passive loss inference in wireless sensor networks based on network coding," Proc. IEEE INFOCOM 2009, pp.1809–1817, Rio de Janeiro, Brazil, April 2009.

[87] J. Liu, D. Goeckel, and D. Towsley, "Bounds on the throughput gain of network coding in unicast and multicast wireless networks," IEEE J. Sel. Areas Commun., vol.27, no.5, pp.582–592, June 2009.

[88] F. Lo Presti, N.G. Duffield, J. Horowitz, and D. Towsley, "Multicast-based inference of network-internal delay distributions," IEEE/ACM Trans. Netw., vol.10, no.6, pp.761–775, Dec. 2002.

[89] D.S. Lun, N. Ratnakar, M. Médard, R. Koetter, D.R. Karger, T. Ho, E. Ahmed, and F. Zhao, "Minimum-cost multicast over coded packet networks," IEEE Trans. Inf. Theory, vol.52, no.6, pp.2608–2623, June 2006

[90] G. Ma, Y. Xu, K. Ou, and W. Luo, "How can network coding help P2P content distribution?," Proc. IEEE ICC 2009, Dresden, Germany, June 2009

[91] E.D. Manley, J.S. Deogun, L. Xu, and D.R. Alexander, "All-optical network coding," IEEE/OSA J. Opt. Commun. and Netw., vol.2, no.4, pp.175–191, April 2010.

[92] Y. Mao, F.R. Kschischang, B. Li, and S. Pasupathy, "A factor graph approach to link loss monitoring in wireless sensor networks," IEEE J. Sel. Areas Commun., vol.23, no.4, pp.820–829, April 2005.

[93] T. Matsuda and T. Takine, "(p,q)-epidemic routing for sparsely populated mobile ad hoc networks," IEEE J. Sel. Areas Commun., vol.26, no.5, pp.783–793, June 2008.

[94] 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, Oct. 2008.

[95] T. Matsuda and T. Takine, "Multicast communications with Reed Solomon/network joint coding in wireless multihop networks," J. Commun., vol.4, no.11, pp.856–864, Dec. 2009.

[96] R. Matsumoto, "Construction algorithm for network error-correcting codes attaining the Singleton bound," IEICE Trans. Fundamentals, vol.E90-A, no.9, pp.1729–1735, Sept. 2007.

[97] B. Ni, N. Santhapuri, Z. Zhong, and S. Nelakuditi, "Routing with opportunistically coded exchanges in wireless mesh networks," Proc. 2nd Workshop on Wireless Mesh Netw. (WiMesh 2006), pp.157–159, Reston, VA, Sept. 2006.

[98] S.-Y. Ni, Y.-C. Tseng, Y.-S. Chen, and J.-P. Sheu, "The broadcast storm problem in a mobile ad hoc network," Proc. ACM MOBICOM'99, pp.151–162, Seattle, WA, Aug. 1999.

[99] 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.

[100] W. Peng and X.-C. Lu, "On the reduction of broadcast redundancy in mobile ad hoc networks," Proc. ACM MOBIHOC 2000, pp.129–130, Boston, MA, Aug. 2000.

[101] P. Popovski and H. Yomo, "Wireless network coding by amplify-and-forward for bi-directional traffic flows," IEEE Commun. Lett., vol.11, no.1, pp.16–18, Jan. 2007.

[102] J. Price and T. Javidi, "Network coding games with unicast flows," IEEE J. Sel. Areas Commun., vol.26, no.7, pp.1302–1316, Sept. 2008.

[103] A. Qayyum, L. Viennot, and A. Laouiti, "Multipoint relaying for flooding broadcast messages in mobile wireless networks," Proc. 35th Annu. Hawaii Int'l Conf. Sys. Sci. (HICSS' 2002), pp.3866–3875, Big Island, HI, Jan. 2002.

[104] S. Ratnasamy and S. McCanne, "Inference of multicast routing trees and bottleneck bandwidths using end-to-end measurements," Proc. IEEE INFOCOM'99, vol.1, pp.353–360, New York, NY, March 1999.

[105] Y.E. Sagduyu and A. Ephremides, "On joint MAC and network coding in wireless ad hoc networks," IEEE Trans. Inf. Theory, vol.53, no.10, pp.3697–3713, Oct. 2007.

[106] C. Sasaki, A. Tagami, and S. Ano, "Implementation approach for network coding using external devices in IP multicast," Proc. 2010 IEEE/IPSJ Int'l Symp. on Applicat. and the Internet (SAINT 2010), pp.293–296, July 2010.

[107] P. Sattari, A. Markopoulou, and C. Fragouli, "Multiple source multiple destination topology inference using network coding," Proc. Workshop on Netw. Coding, Theory and Applicat. (NETCOD 2009), pp.36–41, Lausanne, Switzerland, June 2009.

[108] B. Scheuermann, W. Hu, and J. Crowcroft, "Near-optimal co-ordinated coding in wireless multihop networks," Proc. Int'l Conf. Emerging Netw. Experiments and Tech. (CoNeXT 2007), New York, NY, Dec. 2007.

[109] S. Sengupta, S. Rayanchu, and S. Banerjee, "An analysis of wireless network coding for unicast sessions: the case for coding-aware routing," Proc. INFOCOM 2007, pp.1028–1036, Anchorage, Alaska, May 2007.

[110] S. Sengupta, S. Rayanchu, and S. Banerjee, "Network coding-aware

routing in wireless networks," IEEE/ACM Trans. Netw., vol.18, no.4, pp.1158–1170, Aug. 2010.

[111] S. Shadbakht and B. Hassibi, "MCMC methods for entropy optimization and nonlinear network coding," Proc. 2010 IEEE Int'l Symp. Inf. Theory (ISIT 2010), pp.2383–2387, Austin, TX, June 2010.

[112] M. Sheng, J. Li, and Y. Shi, "Relative degree adaptive flooding broadcast algorithm for ad hoc networks," IEEE Trans. Broadcast., vol.51, no.2, pp.216–222, June 2005.

[113] M.J. Siavoshani, C. Fragouli, and S. Diggavi, "Subspace properties of randomized network coding," Proc. IEEE 2007 Inf. Theory Workshop on Inf. Theory for Wireless Netw., Lake Tahoe, CA, July 2007.

[114] M.J. Siavoshani, C. Fragouli, and S. Diggavi, "On locating Byzantine attackers," Proc. 4th Workshop on Netw. Coding, Theory and Applicat. (NETCOD 2008), Hong Kong, China, Jan. 2008.

[115] D. Silva, F.R. Kschischang, and R. Koetter, "A rank-metric approach to error control in random network coding," IEEE Trans. Inf. Theory, vol.54, no.9, pp.3951–3967, Sept. 2008.

[116] D. Silva and F.R. Kschischang, "Universal weakly secure network coding," Proc. IEEE Inf. Theory Workshop on Netw. and Inf. Theory, pp.281–285, Volos, Greece, June 2009.

[117] R. Subramanian and F. Fekri, "Throughput performance of network-coded multicast in an intermittently-connected network," Proc. 8th Int'l Symp. Modeling and Optimization in Mobile, Ad Hoc and Wireless Netw. (WiOpt 2010), pp.212–221, Avignon, France, May-June 2010.

[118] A.S. Tanenbaum, Computer Networks, 4th ed., Prentice Hall, Upper Saddle River, NJ, 2002.

[119] G.K. Tran, K. Sakaguchi, F. Ono, and K. Araki, "2D MIMO network coding with inter-route interference cancellation," IEICE Trans. Commun., vol.E92-B, no.12, pp.3665–3675, Dec. 2009.

[120] Y. Tsang, M. Coates, and R.D. Nowak, "Network delay tomography," IEEE Trans. Signal Process., vol.51, no.8, pp.2125–2136, Aug. 2003.

[121] M. Tsuru, T. Takine, and Y. Oie, "Inferring link loss rates from unicast-based end-to-end measurement," IEICE Trans. Commun., vol.E85-B, no.1, pp.70–78, Jan. 2002.

[122] D. Umehara, T. Hirano, S. Denno, and M. Morikura, "Throughput analysis of wireless relay slotted aloha systems with network coding," Proc. IEEE GLOBECOM 2008, New Orleans, LA, Nov./Dec. 2008.

[123] A. Vahdat and D. Becker, "Epidemic routing for partially-connected ad hoc networks," Duke Tech. Rep. CS-2000-06, 2000.

[124] C.-C. Wang and X. Lin, "Fast resource allocation for network-coded traffic — A coded-feedback approach," Proc. IEEE INFOCOM 2009, pp.2801–2805, Rio de Janeiro, Brazil, April 2009.

[125] T. Wang and G.B. Giannakis, "Complex field network coding for multiuser cooperative communications," IEEE J. Sel. Areas Commun., vol.26, no.3, pp.561–571, April 2008.

[126] J. Widmer and J.-Y. Le Boudec, "Network coding for efficient communication in extreme networks," Proc. 2005 ACM SIGCOMM Workshop on Delay-Tolerant Netw., pp.284–291, Philadelphia, PA, Aug. 2005.

[127] Y. Wu and S.-Y. Kung, "Distributed utility maximization for network coding based multicasting: a shortest path approach," IEEE J. Sel. Areas Commun., vol.24, no.8, pp.1475–1488, Aug. 2006.

[128] Y. Wu, A. Dimakis, and K. Ramchandran, "Deterministic regenerating codes for distributed storage," Proc. 45th Allerton Conf. Commun., Control, and Comput., Monticello, IL, Oct. 2007.

[129] Y. Wu, "Existence and construction of capacity-achieving network codes for distributed storage," IEEE J. Sel. Areas Commun., vol.28, no.2, pp.277–288, Feb. 2010.

[130] Y. Xi and E.M. Yeh, "Distributed algorithms for minimum cost multicast with network coding," IEEE/ACM Trans. Netw., vol.18, no.2, pp.379–392, April 2010.

[131] L. Xiao, T.F. Fuja, J. Kliewer, and D. Costello, "A network coding approach to cooperative diversity," IEEE Trans. Inf. Theory, vol.53, no.10, pp.3714–3722, Oct. 2007.

[132] D. Xu, Z. Bai, A. Waadt, G.H. Bruck, and P. Jung, "Combining MIMO with network coding: A viable means to provide multiplexing and diversity in wireless relay networks," Proc. IEEE ICC 2010, Cape Town, South Africa, May 2010.

[133] M. Yang and Y. Yang, "A linear inter-session network coding scheme for multicast," J. Communications, vol.4, no.11, pp.865–872, Dec. 2009.

[134] M. Yang and Y. Yang, "A hypergraph approach to linear network coding in multicast networks," IEEE Trans. Parallel Distrib. Syst., vol.21, no.7, pp.968–982, July 2010.

[135] S. Yang, C.K. Ngai, and R.W. Yeung, "Construction of linear network codes that achieve a refined Singleton bound," Proc. 2007 IEEE Int'l Symp. Inf. Theory (ISIT 2007), pp.1576–1580, Nice, France, June 2007.

[136] S. Yang and J. Wu, "Efficient broadcasting using network coding and directional antennas in MANETs," IEEE Trans. Parallel Distrib. Syst., vol.21, no.2, pp.148–161, Feb. 2010.

[137] Z. Yang, M. Li, and W. Lou, "R-Code: network coding based reliable broadcast in wireless mesh networks with unreliable links," Proc. IEEE GLOBECOM 2009, Honolulu, HI, Nov./Dec. 2009.

[138] H. Yao, S. Jaggi, and M. Chen, "Network coding tomography for network failures," Proc. IEEE INFOCOM 2010, San Diego, CA, March 2010.

[139] R.W. Yeung, S.-Y.R. Li, N. Cai, and Z. Zhang, "Network coding theory," Found. and Trends in Commun. and Inf. Theory, vol.2, nos.4-5, pp.241–381, 2005.

[140] R.W. Yeung and N. Cai, "Network error correction, part I: Basic concepts and upper bounds," Commun. in Inf. and Sys., vol.6, no.1, pp.19–35, 2006.

[141] R.W. Yeung, Information Theory and Network Coding, Springer, New York, NY, 2008.

[142] Z. Yu, Y. Wei, B. Ramkumar, and Y. Guan, "An efficient signature-based scheme for securing network coding against pollution attacks," Proc. IEEE INFOCOM 2008, pp.2083–2091, Phoenix, AZ, April 2008.

[143] J. Zhang and Q. Zhang, "Cooperative network coding-aware routing for multi-rate wireless networks," Proc. IEEE INFOCOM 2009, pp.181–189, Rio de Janeiro, Brazil, April 2009.

[144] Q. Zhang, Z. Jin, Z. Zhang, and Y. Shu, "Network coding for applications in the delay tolerant network (DTN)," Proc. 5th Int'l Conf. Mobile Ad-hoc and Sensor Netw. (MSN'09), pp.376– 380, Wu Yi Mountain, China, Dec. 2009.

[145] S. Zhang, S.C. Liew, and P.P. Lam, "Hot topic: Physical-layer network coding," Proc. ACM MOBICOM 2006, pp.358–365, Los Angeles, CA, Sept. 2006.

[146] X. Zhang, G. Neglia, J. Kurose, and D. Towsley, "On the benefits of random linear coding for unicast applications in disruption tolerant networks," Proc. 2nd Workshop on Netw. Coding, Theory and Applicat. (NETCOD 2006), Boston, MA, April 2006.

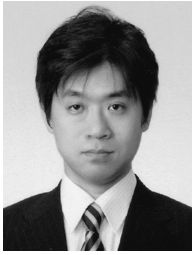
[147] X. Zhang and B. Li, "On the market power of network coding in P2P content distribution systems," Proc. IEEE INFOCOM 2009, pp.334–342, Rio de Janeiro, Brazil, April 2009.

[148] Z. Zhang, "Linear network error correction codes in packet networks," IEEE Trans. Inf. Theory, vol.54, no.1, pp.209–218, Jan. 2008.

[149] F. Zhao, T. Kalker, M. Médard, and K.J. Han, "Signatures for content distribution with network coding," Proc. IEEE Int'l Symp. on Inf. Theory (ISIT 2007), pp.556–560, Nice, France, June 2007.

[150] J. Zhao, F. Yang, Q. Zhang, Z. Zhang, and F. Zhang, "LION: layered overlay multicast with network coding," IEEE Trans. Multimedia, vol.8, no.5, pp.1021–1032, Oct. 2006.

[151] 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, Jan. 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 University. His research interests include performance analysis and the design of communication networks and wireless communications. He is a member of IEEE and IPSJ.

Taku Noguchi received the B.E., M.E. and Ph.D. degrees in communications engineering from Osaka University in 2000, 2002 and 2004, respectively. He joined College of Information Science and Engineering at Ritsumeikan University in 2004, where he is currently an Assistant Professor. His research interests include multicast communication, overlay networks and mobile ad-hoc networks. He is a member of IEEE and IPSJ.

Tetsuya Takine is currently a Professor in 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 2010, 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 IPSJ, ISCIE, and IEEE.