Full-Diversity Binary Frame-Wise Network Coding for Multiple-Source Multiple-Relay Networks Over Slow-Fading Channels

Jun Li, *Member, IEEE*, Jinhong Yuan, *Senior Member, IEEE*, Robert Malaney, *Member, IEEE*, Ming Xiao, *Member, IEEE*, and Wen Chen, *Senior Member, IEEE*

***Abstract*—We study the design of network codes for M-source N-relay ($M - N - 1$) wireless networks over slow-fading channels. Binary frame-wise network coding (BFNC) based on cyclic-shifting matrices is developed to achieve full diversity and good coding gain. We develop a criterion in the context of BFNC that if satisfied guarantees to achieve full diversity gain. Based on this criterion, we propose an algorithm to design low-complexity encoders for a BFNC scheme by exploiting quasi-cyclic low-density parity-check (LDPC) code structures. We also design practical decoders based on the belief propagation decoding principle with a focus on large block lengths. Numerical results demonstrate that our BFNC schemes have substantial benefits over previous complex field and Galois field network coding schemes in the sense that our BFNC schemes can achieve full diversity gain and high coding gain for arbitrary block lengths with low encoding/decoding complexity.**

***Index Terms*—Belief propagation (BP) decoding, low-density parity-check (LDPC) code, multiple-source multiple-relay network, network coding (NC), quasi-cyclic (QC) matrix, slow-fading channel.**

I. Introduction

IN RELAYING networks, one or more intermediate relaying nodes are utilized to help the sources transmit information to the destinations. Relaying networks have been studied for decades [1]. In the last decade, the application of relays has become an efficient technique to combat channel fading and improve system throughput in practical wireless networks [2]–[8]. By allowing information processing in the intermediate nodes, network coding (NC) schemes originally proposed for computer networks have been proved to achieve network multicast capacity bounds [9]. Recently, how to leverage NC in wireless relaying networks to enhance the achievable rates has drawn increasing interest [10]–[16].

Various NC schemes based on the decode-and-forward (DF) protocol have been proposed for multiple-source, multiple-relay, and one-destination wireless systems ($M - N - 1$ relaying networks) to combat slow-fading channels. There are two main classes of NC schemes with DF protocol. One is complex field NC (CFNC) [17], [18], where the frames of the sources are superimposed in a symbol-wise manner at the relay to generate a network-coded frame. Here, the symbol-wise manner means that the ith symbol in the network-coded frame is generated by superimposing the ith symbols of all the sources' frames, and it is independent on the other symbols of the sources. The other type of NC is Galois field NC (GFNC) [19]–[21], where the frames of the sources are operated in a Galois field symbol-wise manner at the relay. Compared with the $M - N - 1$ relaying networks without NC schemes, networks with either the CFNC schemes [17] or the GFNC schemes [19] can achieve higher rates without reducing the full diversity.

However, there are some limitations of the existing CFNC and GFNC schemes. First, in the case that the network size (i.e., M and N) and the frame length (i.e., the number of symbols or Galois filed elements in a frame) are large, we need to choose large space–time matrices (in CFNC) or a large size of the Galois fields (in GFNC). Therefore, the encoding complexity of the CFNC and GFNC schemes will significantly increase. Second, in both CFNC and GFNC schemes, the frames of the sources are network coded in a symbol-wise manner at the relay. Here, symbol-wise manner means that these frames are first aligned and then operated across the ith ($i = 0, \ldots, l$, where l is the frame length) symbols of all the frames. Therefore, there is no connection between the ith symbol of one source and the jth symbol ($j \neq i$) of the other source. Due to these symbol level operations, the destination cannot jointly decode the frames from all the sources and the relays. Hence, when the frame length becomes large, the symbol-wise NC limits the coding gain. Finally, to achieve better coding gain, maximum likelihood (ML) decoding is used in both CFNC and GFNC schemes. However, since the complexity of ML decoding increases according to 2^{Ml}, ML decoding becomes impractical when M and l increase (e.g., $Ml \geq 30$).

Manuscript received August 29, 2011; revised October 30, 2011, December 6, 2011, and January 11, 2012; accepted January 15, 2012. Date of publication January 27. 2012; date of current version March 21, 2012. This work was supported in part by the Australian Research Council Discovery Projects DP110104995, by the Swedish Research Council, and by the National Science Foundation of China under Grant 60972031 and Grant 61161130529. This paper was presented in part at the 2011 International Conference on Communications. The review of this paper was coordinated by Dr. C. Yuen.

J. Li, J. Yuan, and R. Malaney are with the School of Electrical Engineering and Telecommunications, University of New South Wales, Sydney, N.S.W. 2052, Australia (e-mail: jun.li@unsw.edu.au; j.yuan@unsw.edu.au; r.malaney@unsw.edu.au).

M. Xiao is with the ACCESS Linnaeus Center, School of Electrical Engineering, Royal Institute of Technology, 100 44 Stockholm, Sweden (e-mail: ming.xiao@ee.kth.se).

W. Chen is with the Department of Electronic Engineering, Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: wenchen@sjtu.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TVT.2012.2185966

These observations motivate us to consider full diversity binary frame-wise NC (BFNC) schemes with a lower encoding/decoding complexity in $M-N-1$ systems over a slow-fading channel. In the BFNC schemes, the frame-wise operation means that the ith symbol in the network-coded frame can be generated by the binary field addition of arbitrary symbols from all the sources. We note that in [22], full diversity achieving NC schemes are proposed based on root-check (RC) low-density parity-check codes (LDPC) for the 2-1-1 network over slow-fading channels. In the schemes of [22], each network-coded digit at the relay is generated by the binary field addition of arbitrary digits from the sources (according to an RC-LDPC code), leading to full diversity in the 2-1-1 network. However, the schemes of [22] cannot achieve full diversity in the general $M-N-1$ network due to the structure constraint of RC-LDPC codes. This structure constraint is that the parity check matrix must contain a fixed number of identity submatrices to construct the RCs needed to protect the information bits from the fading channel. Due to this constraint, we cannot generate enough RCs to protect every information bit from the fading channel in the general $M-N-1$ network. In [23], LDPC-based NC schemes are designed for the $M-N-1$ network with fast-fading channels. The NC schemes in [23] are designed to achieve a high coding gain. However, [23] demonstrates that the network schemes are not designed to achieve the full diversity of the network.

Based on the preceding discussions, we can see that hitherto there exists no binary code design that achieves full diversity in a general $M-N-1$ network while still achieving high coding gain and low encoding/decoding complexity with large code length. It is the design of such a code that we address in this paper. Our key insight is the design of NC schemes implemented using quasi-cyclic (QC) LDPC codes with the additional constraint that the parity check matrices of the QC-LDPC codes are designed to achieve full diversity for a general $M-N-1$ network with arbitrary block lengths. We note that the QC vector can be seen as a symbol of Galois field, and therefore, it seems that our BFNC can be derived from GFNC. However, the GFNC schemes are not optimized for belief propagation (BP) decoding. In addition, due to the complexity constraint, it is very difficult to jointly decode (by ML decoding) a long codeword based on GFNC. On the other hand, by using our algorithms, we can optimize the degree distributions of our BFNC schemes, which we will show leads to high coding gain, low complexity, and full diversity under BP decoding.

In this paper, we first investigate the full diversity-achieving criterion by treating the BFNC schemes as frame-wise cyclic-shift channel codes. We prove that the criterion can be applied to BFNC schemes with both ML and BP decoding. Then, based on the criterion, we propose an algorithm to design the low-complexity encoders of BFNC schemes by exploiting the parity check matrices of QC-LDPC codes, i.e., QC matrices. Finally, we focus on large block lengths and design practical decoders based on the BP decoding principle. Our codes have the following advantages: 1) The proposed BFNC schemes can be linearly encoded and easily extended to large block lengths. 2) In terms of decoding performance and complexity, for small block lengths, the ML decoder can be used to achieve full diversity. For large block lengths, the proposed BFNC schemes can still achieve full diversity and good coding gain due to the joint decoding of the frames from the sources and the relays by a modified BP decoder (MBP) composed of two concatenated BP decoders.

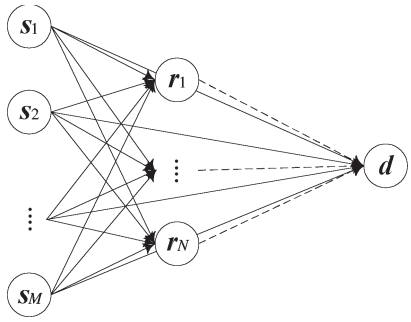


Fig. 1. System model of the $M-N-1$ relaying network.

This paper is organized as follows. Section II sets up the system model. Section III explores the design criterion for the BFNC schemes that can achieve full diversity with either the ML decoder or the BP decoder. Section IV investigates the efficient encoding BFNC schemes based on the QC-LDPC code structures according to the criterions. Section V proposes an MBP that is efficient for large block lengths. Section VI provides the simulation results.

II. System Model and Preliminaries

Fig. 1 shows an $M-N-1$ relaying network, where M sources, i.e., $s_1,\ldots,s_M$, transmit the information frames to their common destination d with the help of N half-duplexing relays $r_1,\ldots,r_N$. Suppose that all transmitting nodes access the channels using time-division multiple access and all the channels are of frequency-nonselective slow fading. During a block of length l_{block}, there are two transmission phases. In the first phase, the sources $s_1,\ldots,s_M$ take turns to broadcast their frames to the relays and the destination with the frame length of l symbols. Note that inclusion of channel coding at the sources will lead to additional coding gain at the cost of additional complexity. Such additional coding can be included in our schemes in a straightforward manner. However, note that we do not consider the joint optimization of channel codes and NC in this paper (see [24] and [25] for more details on the joint design of channel codes and NC). Our main focus is to design network codes to provide both network diversity and coding gain. Therefore, we will not consider channel coding at sources. Each relay tries to decode the information of all sources and encode the sources' information to parity check frames by an NC scheme. Here, we assume that each relay generates a parity check frame of length l. In the second phase, the relays $r_1,\ldots,r_N$ take turns to forward the network-coded frames to the destination, and all sources keep silent. Thus, the whole block length is $l_{\text{block}}=(M+N)l$. After the second phase, the destination decodes the information of the sources by combining the received frames of the two phases.

In [19], it is shown that a GFNC scheme can first be designed to achieve full diversity under the assumption that the source-to-relay channels are error free and the source-to-destination and relay-to-destination channels are Rayleigh distributed. Then, they proved that this GFNC scheme can still achieve full diversity when the source-to-relay channels are changed to Rayleigh fading channels by analyzing various error patterns in the source-to-relay channels. Similarly, in [17], it is shown that the design methodology of CFNC schemes can also follow the same way. In this paper, we design the full diversity-achieving BFNC schemes based on the assumption that the source-to-relay channels are error free. Note that this assumption does not change the NC design method in achieving full diversity of the proposed coding scheme, which is verified both in our simulations and in the analysis of [19]. Specifically, if a relay can decode only some sources' messages, only these decoded messages participate in the NC strategy. We denote the channel coefficient between the mth source s_m, $m = 1, \ldots, M$ and the destination as $\hbar_m$, and the channel coefficient between the nth relay r_n, $n = 1, \ldots, N$ and the destination as h_n. We assume that all channel magnitudes $|\hbar_m|$ and $|h_n|$ are independent identically distributed (i.i.d.) Rayleigh random variables with zero mean and unit variance. All channel coefficients are randomly distributed, but they remain constant for at least one block length $l_{\text{block}} = (M + N)l$.

The frames of all transmitting nodes are binary phase-shift keying (BPSK) modulated. We denote $\boldsymbol{b}_{s_m} = [b_{s_m,1}, \ldots, b_{s_m,l}]^T$ as the information bit vector of s_m and $\boldsymbol{x}_{s_m} = [x_{s_m,1}, \ldots, x_{s_m,l}]^T$ as the transmitting frame after modulation, where the superscript T represents the transpose of a vector. All symbols transmitted by each source are uniform i.i.d. We denote $\boldsymbol{b}_{r_n} = [b_{r_n,1}, \ldots, b_{r_n,l_r}]^T$ as the network-coded bit vector of r_n, which is generated based on the frames from the sources by a BFNC scheme. Correspondingly, the modulated frame transmitted by r_n is $\boldsymbol{x}_{r_n} = [x_{r_n,1}, \ldots, x_{r_n,l_r}]^T$. In the BPSK modulation, we have $\boldsymbol{x}_{s_m} = (-1)^{\boldsymbol{b}_{s_m}}$ and $\boldsymbol{x}_{r_n} = (-1)^{\boldsymbol{b}_{r_n}}$. We suppose that the average power of the transmitted symbols at both the sources and the relays is the same and denoted as P. The additive channel noise at the destination receiver is i.i.d. Gaussian distributed with variance σ^2. Therefore, the average signal-to-noise ratio is defined as $\rho \triangleq P/\sigma^2$. During a block period, we denote $\boldsymbol{y}_{1m}$ as the received frame from s_m in the first phase and $\boldsymbol{y}_{2n}$ as the received frame from r_n in the second phase. Then, we have $\boldsymbol{y}_{1m} = \hbar_m \boldsymbol{x}_{s_m} + \boldsymbol{v}_{1m}$ and $\boldsymbol{y}_{2n} = h_n \boldsymbol{x}_{r_n} + \boldsymbol{v}_{2n}$, where $\boldsymbol{v}_{1m}$ and $\boldsymbol{v}_{2n}$ are the two vectors of noise samples.

We investigate the instantaneous mutual information of the network in a block period. Since the source-to-relay channels are assumed error free, the minimum cut-set of the network is determined by the source-to-destination channels and the relay-to-destination channels. We calculate the mutual information based on Gaussian signaling. Note that we use Gaussian signals to derive a lower bound on the outage probability for BPSK. This lower bound leads to the same diversity gain achieved using BPSK. We focus on the mth source s_m. The mutual information between s_m and the destination on the s_m-to-d channel is $\log(1 + |\hbar_m|^2\rho)$. In addition, according to the outage probability analysis in [18], the information transmitted by the N relays can be averaged on the M sources. This is because the NC at the relays is randomly constructed. Then, the mutual information for s_m, which is contributed by the relays, is written as $(1/M)\sum_{n=1}^{N} \log(1 + |h_n|^2\rho)$. The mutual information between s_m and the destination can be written as

$$I_{BFNC}^{s_m} = \frac{1}{1 + \frac{N}{M}} \times \left(\log\left(1 + |\hbar_m|^2\rho\right) + \frac{1}{M}\sum_{n=1}^{N} \log\left(1 + |h_n|^2\rho\right) \right). \quad (1)$$

Note that in (1), $1 + N/M$ in the denominator represents the effective number of frames allocated to the source s_m, where N/M represents the effective number of frames transmitted by the relays for s_m.

In the BFNC schemes, the network-coded frames transmitted by the N relays can be seen as the parity check frames for the M sources. Therefore, the transmission rate of each block is $R = M/(M + N)$. Since the N parity check frames are shared by the M sources, the transmission rate of each source keeps the same value R. Given the transmission rate R and the channel realization $\boldsymbol{h} = [\hbar_1, \ldots, \hbar_M, h_1, \ldots, h_N]^T$ in a block period, the outage probability of s_m is defined as $P_r(\mathcal{O}_{s_m}) \triangleq P_r(I_{BFNC}^{s_m} < R|\boldsymbol{h})$, where $\mathcal{O}_{s_m}$ represents the outage event of s_m. Thus, the outage probability of the network can be calculated as $P_o = P_r(\mathcal{O}_{s_1} \cup \cdots \cup \mathcal{O}_{s_M})$, which can be seen as the lower bound of the block error probability (BLEP) P_e. We define the diversity gain as $\lambda \triangleq -\lim_{\rho\to\infty} \log P_e / \log\rho$. According to [19], since the information of each source is transmitted in $N + 1$ independent channels to the destination, the maximum diversity gain of the $M - N - 1$ network is $\lambda = N + 1$.

III. Full Diversity Achieving Criterion for Binary Frame-Wise Network Coding Schemes

In this section, we will design the BFNC schemes that can achieve full diversity with either the ML decoder or the BP decoder at the destination.

A. BFNC Design Criterion With ML Decoder

Let us denote by $\boldsymbol{x}$ the signal frames transmitted by all sources and the relays. Therefore, we have $\boldsymbol{x} = [\boldsymbol{x}_{s_1}^T, \ldots, \boldsymbol{x}_{s_M}^T, \boldsymbol{x}_{r_1}^T, \ldots, \boldsymbol{x}_{r_N}^T]^T$. Note that in [26], the bounds of BLEP of a network-coded multiuser system have been well developed. However, in this paper, we mainly focus on the code design to achieve full diversity gain. Therefore, we study the pairwise error probability (PEP), i.e., $P(\boldsymbol{x} \to \hat{\boldsymbol{x}})$, of the network, which is defined as the average error probability of the event that a block $\boldsymbol{x}$ is decoded into another block $\hat{\boldsymbol{x}} = [\hat{\boldsymbol{x}}_{s_1}^T, \ldots, \hat{\boldsymbol{x}}_{s_M}^T, \hat{\boldsymbol{x}}_{r_1}^T, \ldots, \hat{\boldsymbol{x}}_{r_N}^T]^T$ with the ML decoder. To investigate the diversity gain of the $M - N - 1$ relaying network, we first derive the expression of PEP shown in the following lemma.

Lemma 1: When ρ is large enough, the PEP of the $M-N-1$ relaying network with ML decoder is

$$P(\boldsymbol{x}\to\hat{\boldsymbol{x}}) = \frac{1}{2} \cdot \frac{1\cdot 3\cdots(2M+2N-1)\cdot 2^{3(M+N)}}{2\cdot 4\cdots(2M+2N)\prod_{m=1}^{M}(1+\|\boldsymbol{u}_{s_m}\|^2\rho)\cdot\prod_{n=1}^{N}(1+\|\boldsymbol{u}_{r_n}\|^2\rho)} \tag{2}$$

where $\boldsymbol{u}_{s_m} = 1/\sqrt{P}(\boldsymbol{x}_{s_m} - \hat{\boldsymbol{x}}_{s_m})$, $m = 1,\ldots,M$, and $\boldsymbol{u}_{r_n} = 1/\sqrt{P}(\boldsymbol{x}_{r_n} - \hat{\boldsymbol{x}}_{r_n})$, $n = 1,\ldots,N$. The Frobenius two-norm $\|\boldsymbol{z}\|$ of a vector $\boldsymbol{z} = [z_1,\ldots,z_l]^T$ is calculated as $\|\boldsymbol{z}\| = \sqrt{z_1^2+\cdots+z_l^2}$.

Proof: See Appendix A.

We define the frame-wise Hamming distance of two blocks $\boldsymbol{x}$ and $\hat{\boldsymbol{x}}$ as the number of different frames (e.g., $\boldsymbol{x}_{s_m} \neq \hat{\boldsymbol{x}}_{s_m}$ or $\boldsymbol{x}_{r_n} \neq \hat{\boldsymbol{x}}_{r_n}$) between the two blocks. From Lemma 1, the diversity gain of the $M-N-1$ relaying network is determined by the minimum frame-wise Hamming distance of arbitrary two blocks $\boldsymbol{x}$ and $\hat{\boldsymbol{x}}$. For a conventional bit-wise BFNC scheme [10], since each network coded bit at a relay is XORed in bit-wise fashion, all relays generate the same network-coded frame as $\sum_{m=1}^{M}\oplus\boldsymbol{b}_{s_m} = \boldsymbol{b}_{s_1}\oplus\boldsymbol{b}_{s_2}\oplus\cdots\oplus\boldsymbol{b}_{s_M}$, where $\sum\oplus$ represents the XOR operation among multiple bit frames. We can see that in the conventional bit-wise BFNC scheme, the minimum frame-wise Hamming distance of any two blocks is two. Therefore, the diversity gain of the conventional BFNC scheme in the $M-N-1$ relaying network is two regardless the number of relays in the network. To achieve full diversity, i.e., $(N+1)$-order diversity, in the $M-N-1$ relaying network, we consider the frame-wise BFNC schemes. In these schemes, the network-coded frame at the relay r_n, i.e., $\boldsymbol{b}_{r_n}$, is generated to satisfy $\boldsymbol{H}_{r_n}\boldsymbol{b}_{r_n} = \sum_{m=1}^{M}\oplus\boldsymbol{H}_{s_m,n}\boldsymbol{b}_{s_m}$, where $\boldsymbol{H}_{r_n}$, $\boldsymbol{H}_{s_1,n},\ldots,\boldsymbol{H}_{s_M,n}$ are all $l\times l$ binary matrices corresponding to parity-check submatrices for the network codeword at the relay r_n. Let us define the entire network codeword as $\boldsymbol{b} = [\boldsymbol{b}_{s_1}^T,\ldots,\boldsymbol{b}_{s_M}^T,\boldsymbol{b}_{r_1}^T,\ldots,\boldsymbol{b}_{r_N}^T]^T$. The parity check matrix of the network code for the M sources and the N relays with frame length l for each transmission is denoted by an $Nl\times(M+N)l$ binary matrix $\boldsymbol{H}$ given by

$$\boldsymbol{H} = \begin{bmatrix} \boldsymbol{H}_{s_1,1} & \boldsymbol{H}_{s_2,1} & \cdots & \boldsymbol{H}_{s_M,1} & \boldsymbol{H}_{r_1} & \boldsymbol{O} & \cdots & \boldsymbol{O} \\ \boldsymbol{H}_{s_1,2} & \boldsymbol{H}_{s_2,2} & \cdots & \boldsymbol{H}_{s_M,2} & \boldsymbol{O} & \boldsymbol{H}_{r_2} & \cdots & \boldsymbol{O} \\ & & \cdots & & & & \cdots & \\ \boldsymbol{H}_{s_1,N} & \boldsymbol{H}_{s_2,N} & \cdots & \boldsymbol{H}_{s_M,N} & \boldsymbol{O} & \boldsymbol{O} & \cdots & \boldsymbol{H}_{r_N} \end{bmatrix} \tag{3}$$

with $\boldsymbol{O}$ representing the $l\times l$ zero matrix. Each BFNC scheme corresponds to a particular binary matrix $\boldsymbol{H}$ in (3). We call matrix $\boldsymbol{H}$ the parity check matrix of a BFNC scheme. Obviously, we have $\boldsymbol{H}\boldsymbol{b} = \boldsymbol{o}$, where $\boldsymbol{o}$ represents the zero vector of length l_{block}.

We rewrite $\boldsymbol{H}$ in (3) as

$$\boldsymbol{H} = [\boldsymbol{H}_1,\ldots,\boldsymbol{H}_M,\boldsymbol{H}_{M+1},\ldots,\boldsymbol{H}_{M+N}] \tag{4}$$

where $\boldsymbol{H}_m = [\boldsymbol{H}_{s_m,1},\ldots,\boldsymbol{H}_{s_m,N}]^T$, $m = 1,\ldots,M$, and $\boldsymbol{H}_{M+1} = [\boldsymbol{H}_{r_1},\boldsymbol{O},\ldots,\boldsymbol{O}]^T,\ldots,\boldsymbol{H}_{M+N} = [\boldsymbol{O},\boldsymbol{O},\ldots,\boldsymbol{H}_{r_N}]^T$. The size of each matrix of $\boldsymbol{H}_1,\ldots,\boldsymbol{H}_M,\boldsymbol{H}_{M+1},\ldots,\boldsymbol{H}_{M+N}$ is $Nl\times l$. Consider a matrix $\boldsymbol{A}$ that is constructed in the following manner: We randomly choose N matrices, e.g., $\boldsymbol{H}_{(1)},\boldsymbol{H}_{(2)},\ldots,\boldsymbol{H}_{(N)}$, from the $M+N$ matrices $\boldsymbol{H}_1,\ldots,\boldsymbol{H}_{M+N}$. We construct $\boldsymbol{A}$ based on the N selected matrices as $\boldsymbol{A} = [\boldsymbol{H}_{(1)},\boldsymbol{H}_{(2)},\ldots,\boldsymbol{H}_{(N)}]$. Therefore, $\boldsymbol{A}$ is a square matrix, and its size is $Nl\times Nl$. The following theorem provides a design criterion of $\boldsymbol{H}$, by which the BFNC scheme can achieve $(N+1)$-order diversity with the ML decoder.

Theorem 1: The $M-N-1$ relaying network with ML decoder can achieve $(N+1)$-order diversity if all the columns of the matrix $\boldsymbol{A}$ are linearly independent.

Proof: See Appendix B.

Another way we state Theorem 1 is the $M-N-1$ relaying network with ML decoder can achieve $(N+1)$-order diversity if all the columns in any arbitrary N matrices selected from $\boldsymbol{H}_1,\ldots,\boldsymbol{H}_{M+N}$ are linearly independent. Theorem 1 provides a design criterion for the full diversity-achieving BFNC schemes with ML decoders. In the following, we call this criterion the linearly independent criterion (LIC).

B. BFNC Design Criterion With BP Decoder

When the block length becomes large, the ML decoder has a high complexity. To reduce the decoding complexity, we resort to iterative BP decoding algorithms. In our BFNC schemes, we may also design $\boldsymbol{H}$ in (3) as an LDPC matrix that satisfies the LIC. In this case, iterative BP decoding can be applied to recover the information from the M sources. We now design the full diversity-achieving BFNC schemes for the $M-N-1$ relaying network with a BP decoder. We design the BFNC schemes by borrowing the concept of RC-LDPC codes. Recall that the RC-LDPC codes are designed to achieve full diversity for block fading channels [22]. To achieve full diversity under BP decoder, RC-LDPC codes are intentionally constructed to ensure that each information digit is connected (by the RCs) to the digits transmitted from all the other channels. In such a way, all the information bits can receive extrinsic mutual information from other channels under BP decoding and thus achieve full diversity. This process is known as diversity evolution [22].

Note that 1) to achieve $(N+1)$-order diversity in the network, the destination needs to recover the source's frames in situations where N channels are in deep fading, and 2) the destination knows the coefficients of all M source-to-destination channels and all N relay-to-destination channels by channel estimation. Accordingly, we separate the $M+N$ channels as two groups. One group is composed of M good channels, i.e., M channels with the highest channel gains, and the other group is composed of N bad channels, i.e., N channels with the lowest channel gains. Hopefully, the information digits of N bad channels are root checked by M good channels. Without loss of generality, let us consider a channel realization that the matrices $\boldsymbol{H}_1,\ldots,\boldsymbol{H}_N$ are related to the N frames (denoted as $\boldsymbol{b}_1,\ldots,\boldsymbol{b}_N$) that are transmitted through the N bad channels. We also suppose that the matrices $\boldsymbol{H}_{N+1},\ldots,\boldsymbol{H}_{N+M}$ are related to the M bit frames (denoted as $\boldsymbol{b}_{N+1},\ldots,\boldsymbol{b}_{N+M}$) that are transmitted through the M good channels. If the columns

in $\boldsymbol{H}_1, \ldots, \boldsymbol{H}_N$ are constructed to be linearly independent, we can transfer the matrix $\boldsymbol{H}$ to $\boldsymbol{H}' = [\boldsymbol{I}_{Nl\times Nl}\widetilde{\boldsymbol{H}}]$ by Gaussian elimination, where $\boldsymbol{I}_{Nl\times Nl}$ is an $Nl \times Nl$ identity matrix, and $\widetilde{\boldsymbol{H}}$ is an $Nl \times Ml$ binary matrix. Since $\boldsymbol{H}'$ is derived from $\boldsymbol{H}$ by Gaussian elimination, both $\boldsymbol{H}$ and $\boldsymbol{H}'$ are the parity check matrices of the transmitted bit blocks such that $\boldsymbol{H}\boldsymbol{b} = \boldsymbol{H}'\boldsymbol{b} = \boldsymbol{o}$.

According to the concept of RC-LDPC codes, in the parity check matrix $\boldsymbol{H}'$, all bits transmitted in the N bad channels (corresponding to the identical matrix $\boldsymbol{I}$ in $\boldsymbol{H}'$) can obtain the extrinsic mutual information from the M good channels (corresponding to the submatrix $\widetilde{\boldsymbol{H}}$ in $\boldsymbol{H}'$) with one iteration of the BP decoder. Thus, all bits in a block are equivalently transmitted in the M good channels, and the network code achieves $(N+1)$-order diversity based on the diversity evolution of the RC-LDPC codes. Here, we call $\boldsymbol{H}'$ the RC-LDPC matrix for this channel realization. Note that the N bad channels are randomly distributed in all the source-to-destination channels and all the relay-to-destination channels for different transmission blocks. Therefore, to achieve full diversity with a BP decoder, all the columns in any arbitrary N matrices selected from $\boldsymbol{H}_1, \ldots, \boldsymbol{H}_{M+N}$ should be linearly independent. For different channel realizations, we need to obtain different RC-LDPC matrices $\boldsymbol{H}'$ at the destination for the BP decoder to obtain $(N+1)$-order diversity.

We note that the criterion to achieve full diversity with the ML and the BP decoding are the same, i.e., LIC can be utilized to design the diversity-achieving BFNC schemes with either the ML or the BP decoders. For the BFNC schemes with BP decoders, we need to transfer the parity check matrix $\boldsymbol{H}$ according to the channel conditions to obtain the RC-LDPC structure $\boldsymbol{H}'$. In Section V, we will discuss in detail how to design MBPs to achieve full diversity and good coding gain.

IV. Low-Complexity Binary Frame-Wise Network Coding Encoder Design Based on Quasi-Cyclic Matrices

From an encoding complexity perspective, complex field operations in the CFNC schemes and Galois field operations in the GFNC schemes are utilized to generate network-coded symbols at the relays. These types of operations are of high encoding complexity if M, N, and frame length l are large. Specifically, in a CFNC scheme, there are Ml complex field multiplication operations and $(M-1)l$ complex field addition operations to generate a network-coded frame at each relay. Therefore, there are in total MNl complex field multiplication operations and $(M-1)Nl$ complex field addition operations for the encoding process of a CFNC scheme. In a GFNC scheme, if we view a frame as a symbol from GF (2^q) $(1 \le q \le l)$, then there are in total $MN(l/q)$ multiplication operations and $(M-1)N(l/q)$ addition operations over GF (2^q).

For BFNC, we will design the matrix $\boldsymbol{H}$ based on the parity check matrices of QC-LDPC codes, i.e., QC matrices, to satisfy the LIC. The reasons that we choose QC matrices are as follows. First, QC matrices are half deterministic. Therefore, we can design the matrices that satisfy the LIC rather than via an exhaustive search. Second, QC-LDPC code structures are binary and enable linear encoding at the relays, which offers a low-complexity encoding solution. A QC matrix is composed of submatrices that are either zero matrices or circulant permutation matrices. Since QC matrices are designed to be sparse, the encoding process of a BFNC scheme averagely has MNl multiplication operations and $(M-1)Nl$ XOR operations over the binary field. We can see that the encoding complexity of BFNC is less than that of CFNC. In addition, note that one multiplication operation over GF (2^q) is generally composed of (q^3+q^2) multiplication operations and (q^3-q) XOR operations over the binary field, and one addition operation over GF (2^q) corresponds to q XOR operations over the binary field [28]. Thus, in a GFNC scheme, there are $MNl(q^2+q)$ multiplication operations and $MNl(q^2-1)+(M-1)Nl = MNlq^2 - Nl$ XOR operations over the binary field. Compared with the GFNC schemes, the BFNC schemes have a lower encoding complexity.

According to [29], we denote $\boldsymbol{I}^l(\alpha)$ as the circulant permutation matrix that circularly shifts the $l \times l$ identity matrix to the right by α times for any non-negative integer α. For example, if we set $l = 3$ and $\alpha = 2$, then $\boldsymbol{I}^3(2) = [0\,0\,1; 1\,0\,0; 0\,1\,0]$. In addition, we denote $\boldsymbol{I}^l(0)$ as an $l \times l$ identity matrix and $\boldsymbol{I}^l(\infty)$ as an $l \times l$ zero matrix. To turn the matrix $\boldsymbol{H}$ to a QC matrix, we replace its submatrices, e.g., $\boldsymbol{H}_{s_m,n}$, $\boldsymbol{H}_{r_n}$, in (3) with either circulant permutation matrices or square QC matrices. If we put the circulant permutation matrices into the matrix $\boldsymbol{H}$ in (3), then we have $\boldsymbol{H}_{s_m,n} = \boldsymbol{I}^l(\alpha_{nm})$, $\boldsymbol{H}_{r_n} = \boldsymbol{I}^l(\alpha_n)$, and

$$\boldsymbol{H} = \begin{bmatrix} \boldsymbol{I}^l(\alpha_{11}) & \cdots & \boldsymbol{I}^l(\alpha_{1M}) & \boldsymbol{I}^l(\alpha_1) & \boldsymbol{I}^l(\infty) & \cdots & \boldsymbol{I}^l(\infty) \\ \boldsymbol{I}^l(\alpha_{21}) & \cdots & \boldsymbol{I}^l(\alpha_{2M}) & \boldsymbol{I}^l(\infty) & \boldsymbol{I}^l(\alpha_2) & \cdots & \boldsymbol{I}^l(\infty) \\ & \cdots & & & & \cdots & \\ \boldsymbol{I}^l(\alpha_{N1}) & \cdots & \boldsymbol{I}^l(\alpha_{NM}) & \boldsymbol{I}^l(\infty) & \boldsymbol{I}^l(\infty) & \cdots & \boldsymbol{I}^l(\alpha_N) \end{bmatrix}. \tag{5}$$

In (5), the numbers α_n are positive integers, and α_{nm} are either positive integers or ∞. In addition, we can set $\boldsymbol{H}_{s_m,n}$ and $\boldsymbol{H}_{r_n}$ as the square QC matrices. For example, $\boldsymbol{H}_{s_m,n}$ can be composed of a combination of four $l/2 \times l/2$ circulant permutation matrices or $l/2 \times l/2$ zero matrices such that $\boldsymbol{H}_{s_m,n} = [\boldsymbol{I}^{l/2}(0)\boldsymbol{I}^{l/2}(1); \boldsymbol{I}^{l/2}(2)\boldsymbol{I}^{l/2}(\infty)]$. In general, $\boldsymbol{H}_{s_m,n}$ can be composed of a total k^2 of $l/k \times l/k$ circulant permutation matrices, where l is an integer multiple of k.

In matrix $\boldsymbol{H}$, by replacing a zero matrix with 0 and replacing a non-zero circulant matrix with 1, we obtain a basic binary matrix $\hat{\boldsymbol{H}}$. If $\boldsymbol{H}_{s_m,n}$ and $\boldsymbol{H}_{r_n}$ are the circulant permutation matrices, then $\hat{\boldsymbol{H}}$ is an $N \times (M+N)$ matrix. Moreover, if both $\boldsymbol{H}_{s_m,n}$ and $\boldsymbol{H}_{r_n}$ are composed of $k \times k$ circulant permutation matrices of size $l/k \times l/k$, then $\hat{\boldsymbol{H}}$ is an $Nk \times (M+N)k$ matrix. According to [29], if we want to design a matrix $\boldsymbol{H} = [\boldsymbol{H}_1, \ldots, \boldsymbol{H}_{M+N}]$ to satisfy the LIC, a sufficient condition is that we need to design its basic matrix $\hat{\boldsymbol{H}} = [\hat{\boldsymbol{H}}_1, \ldots, \hat{\boldsymbol{H}}_{M+N}]$ so that the columns in arbitrary N of matrices $\hat{\boldsymbol{H}}_1, \ldots, \hat{\boldsymbol{H}}_{M+N}$ are linearly independent. We call this sufficient condition as the basic matrix LIC (BLIC). Therefore, in the QC matrix-based BFNC scheme design, we construct a basic matrix $\hat{\boldsymbol{H}}$

TABLE I
DESCRIPTION OF ALGORITHM 1 THAT IS USED TO GENERATE A SERIES OF FULL COLUMN RANK BINARY MATRICES

Algorithm 1 :

Input: Positive integer υ.
Output: Binary matrices $\boldsymbol{G}_{2,q}$, $q = 1, 2, \cdots, 2^{\upsilon-1}$.
Steps:
1: Choose a positive integer υ ($\upsilon > 2$) such that $f(x) = x^{\upsilon} + x^{\upsilon-1} + 1$ is a primitive polynomial over GF(2) (e.g., υ can be chosen as $3, 4, 6, 7$ etc.).
2: Choose υ linearly independent binary columns $\boldsymbol{b}_1, \cdots, \boldsymbol{b}_{\upsilon}$, with column length equal to or larger than υ.
3: Generate $\boldsymbol{b}_{\upsilon+1}, \boldsymbol{b}_{\upsilon+2}, \cdots$, and $\boldsymbol{b}_{2^{\upsilon}-1}$ by using $f(x) = x^{\upsilon} + x^{\upsilon-1} + 1$ as the generator polynomial, i.e.,

$$\begin{aligned} \boldsymbol{b}_{\upsilon+1} &= \boldsymbol{b}_1 \oplus \boldsymbol{b}_2, \\ \boldsymbol{b}_{\upsilon+2} &= \boldsymbol{b}_2 \oplus \boldsymbol{b}_3, \\ &\vdots \\ \boldsymbol{b}_{2^{\upsilon}-1} &= \boldsymbol{b}_{2^{\upsilon}-1-\upsilon} \oplus \boldsymbol{b}_{2^{\upsilon}-\upsilon}. \end{aligned}$$

4: Obtain $2^{\upsilon} - 1$ binary matrices as $\boldsymbol{G}_{2,1} = [\boldsymbol{b}_1, \boldsymbol{b}_2, \cdots, \boldsymbol{b}_{\upsilon}]$, $\boldsymbol{G}_{2,2} = [\boldsymbol{b}_2, \boldsymbol{b}_3, \cdots, \boldsymbol{b}_{\upsilon+1}]$, $\cdots$, $\boldsymbol{G}_{2,2^{\upsilon}-1} = [\boldsymbol{b}_{2^{\upsilon}-1}, \boldsymbol{b}_1, \cdots, \boldsymbol{b}_{\upsilon-1}]$.

that satisfies the BLIC instead of searching for a matrix $\boldsymbol{H}$ that satisfies the LIC. For simplicity, we first consider the basic matrix design for a $M - 2 - 1$ relaying network. Then, we extend the design to the general $M - N - 1$ relaying network.

A. Basic Matrix Design for the $M - 2 - 1$ Relaying Network

There are two steps to construct a basic matrix $\hat{\boldsymbol{H}} = [\hat{\boldsymbol{H}}_1, \ldots, \hat{\boldsymbol{H}}_{M+2}]$ so that all the columns in any arbitrary two matrices selected from $\hat{\boldsymbol{H}}_1, \ldots, \hat{\boldsymbol{H}}_{M+2}$ are linearly independent. First, we need to make sure that the columns inside each $\hat{\boldsymbol{H}}_k, k = 1, \ldots, M + 2$ are linearly independent, i.e., each matrix $\hat{\boldsymbol{H}}_k$ is of full column rank.

In Table I, we present Algorithm 1 to generate a series of full column rank binary matrices $\boldsymbol{G}_{2,k}$, $k = 1, \ldots, 2^{\upsilon} - 1$, where υ is defined in Algorithm 1. We have two lemmas based on Algorithm 1 as follows.

Lemma 2: Matrices $\boldsymbol{G}_{2,k}$, $k = 1, \ldots, 2^{\upsilon} - 1$, are $2^{\upsilon} - 1$ unique matrices, with the columns in each $\boldsymbol{G}_{2,k}$ being linearly independent.

Proof: See Appendix C.

Lemma 3: The sequence of matrices $\boldsymbol{G}_{2,1}, \ldots, \boldsymbol{G}_{2,2^{\upsilon}-1}$ is closed for addition over GF(2), i.e., for arbitrary i and j ($i, j \in \{1, \ldots, 2^{\upsilon} - 1\}$ and $i \neq j$), there exists $q \in \{1, \ldots, 2^{\upsilon} - 1\}$ such that $\boldsymbol{G}_{2,i} \oplus \boldsymbol{G}_{2,j} = \boldsymbol{G}_{2,q}$.

Proof: See Appendix D.

In Table II we present Algorithm 2 to generate a basic matrix $\hat{\boldsymbol{H}}$ based on Algorithm 1.

Theorem 2: If we construct a basic matrix $\hat{\boldsymbol{H}} = [\hat{\boldsymbol{H}}_1, \ldots, \hat{\boldsymbol{H}}_{M+2}]$ by Algorithm 2, then all the columns in every two matrices selected from $\hat{\boldsymbol{H}}_1, \ldots, \hat{\boldsymbol{H}}_{M+2}$ are linearly independent, i.e., the basic matrix $\hat{\boldsymbol{H}}$ satisfies the BLIC.

Proof: See Appendix E.

By using Algorithm 2, we can generate a basic matrix as $\hat{\boldsymbol{H}} = [\hat{\boldsymbol{H}}_1, \ldots, \hat{\boldsymbol{H}}_{2^{\upsilon}+1}]$, where $2^{\upsilon} + 1 = M + 2$. That is, Algorithm 2 can support an $M - 2 - 1$ relaying network with $M \leq 2^{\upsilon} - 1$ to achieve full diversity. We use the following example to illustrate Algorithm 2. We choose $\upsilon = 3$, and then,

TABLE II
DESCRIPTION OF ALGORITHM 2 THAT IS USED TO GENERATE BASIC MATRICES FOR M-SOURCE, TWO-RELAY NETWORKS

Algorithm 2 :

Input: Positive integer υ.
Output: Binary matrices $\hat{\boldsymbol{H}}_q$, $q = 1, 2, \cdots, 2^{\upsilon} + 1$.
Steps:
1: Choose a positive integer υ ($\upsilon > 2$) such that $f(x) = x^{\upsilon} + x^{\upsilon-1} + 1$ is a primitive polynomial over GF(2).
2: Choose 2υ linearly independent binary columns of length 2υ, e.g., columns from the $2\upsilon \times 2\upsilon$ identity matrix.
3: Split the 2υ columns into two matrices $\boldsymbol{G}_1$ and $\boldsymbol{G}_2$, with υ columns in each matrix.
4: Generate $2^{\upsilon} - 1$ matrices $\boldsymbol{G}_{2,1}, \cdots, \boldsymbol{G}_{2,2^{\upsilon}-1}$ based on the columns in $\boldsymbol{G}_2$ according to **Algorithm** 1. Set $\boldsymbol{G}_{2,1} = \boldsymbol{G}_2$.
5: Generate another $2^{\upsilon} - 1$ matrices by

$$\begin{aligned} \boldsymbol{G}_3 &= \boldsymbol{G}_1 \oplus \boldsymbol{G}_{2,1} \\ \boldsymbol{G}_4 &= \boldsymbol{G}_1 \oplus \boldsymbol{G}_{2,2} \\ &\vdots \\ \boldsymbol{G}_{2^{\upsilon}+1} &= \boldsymbol{G}_1 \oplus \boldsymbol{G}_{2,2^{\upsilon}-1}. \end{aligned}$$

6: Randomly choose j ($j = 2, \cdots, \upsilon$) columns of the υ columns in $\boldsymbol{G}_k$, $k = 1, \cdots, 2^{\upsilon} + 1$, and combine them with a XOR operation to generate a new column. There are a total $\sum_{j=2}^{\upsilon} C_{\upsilon}^{j} = 2^{\upsilon} - \upsilon - 1$ new columns based on the matrix $\boldsymbol{G}_k$. Together with the original υ columns in $\boldsymbol{G}_k$, we obtain a new matrix $\boldsymbol{B}_k$ composed of these $2^{\upsilon} - 1$ columns.
7: Obtain $2^{\upsilon} + 1$ matrices $\boldsymbol{B}_k$, $k = 1, \cdots, 2^{\upsilon} + 1$.
8: Within each matrix $\boldsymbol{B}_k$, randomly choose υ linearly independent columns to construct $\hat{\boldsymbol{H}}_k$.

each submatrix in (3) is composed of nine $l/3 \times l/3$ circulant permutation matrices or an $l/3 \times l/3$ zero matrix. Here, $\boldsymbol{H}_{r_n}$ and $\boldsymbol{H}_{s_m,n}$ in (3) can be written as

$$\boldsymbol{H}_{r_n} = \begin{bmatrix} \boldsymbol{I}^{l/3}(\alpha_{n,1}) & \boldsymbol{I}^{l/3}(\alpha_{n,2}) & \boldsymbol{I}^{l/3}(\alpha_{n,3}) \\ \boldsymbol{I}^{l/3}(\alpha_{n,4}) & \boldsymbol{I}^{l/3}(\alpha_{n,5}) & \boldsymbol{I}^{l/3}(\alpha_{n,6}) \\ \boldsymbol{I}^{l/3}(\alpha_{n,7}) & \boldsymbol{I}^{l/3}(\alpha_{n,8}) & \boldsymbol{I}^{l/3}(\alpha_{n,9}) \end{bmatrix}$$

$$\boldsymbol{H}_{s_m,n} = \begin{bmatrix} \boldsymbol{I}^{l/3}(\alpha_{nm,1}) & \boldsymbol{I}^{l/3}(\alpha_{nm,2}) & \boldsymbol{I}^{l/3}(\alpha_{nm,3}) \\ \boldsymbol{I}^{l/3}(\alpha_{nm,4}) & \boldsymbol{I}^{l/3}(\alpha_{nm,5}) & \boldsymbol{I}^{l/3}(\alpha_{nm,6}) \\ \boldsymbol{I}^{l/3}(\alpha_{nm,7}) & \boldsymbol{I}^{l/3}(\alpha_{nm,8}) & \boldsymbol{I}^{l/3}(\alpha_{nm,9}) \end{bmatrix} \quad (6)$$

where α with different subscripts represent non-negative integers. First, we construct the following $2^{\upsilon} + 1 = 9$ binary matrices based on Algorithm 2. These nine binary matrices $\boldsymbol{B}_k$, $k = 1, \ldots, 9$ are shown in Fig. 2. Then, we construct $\hat{\boldsymbol{H}}$ by obtaining $\hat{\boldsymbol{H}}_k$ from $\boldsymbol{B}_k$. We obtain $\hat{\boldsymbol{H}}_k$ by randomly choosing three linearly independent columns of $\boldsymbol{B}_k$. Therefore, the columns in any two different matrices of $\hat{\boldsymbol{H}}_1, \ldots, \hat{\boldsymbol{H}}_9$ are linearly independent, and $\hat{\boldsymbol{H}}$ satisfies the BLIC. We can see that by choosing $\upsilon = 3$, Algorithm 2 can support the networks to achieve full diversity (third order) when $M + 2 \leq 9$, i.e., $M \leq 7$.

B. Basic Matrix Design for the $M - N - 1$ Relaying Network

We now consider general $M - N - 1$ networks with $M > 2$. Note that in the $M - N - 1$ relaying network, the BLIC for the basic matrix $\hat{\boldsymbol{H}} = [\hat{\boldsymbol{H}}_1, \ldots, \hat{\boldsymbol{H}}_{M+N}]$ is that all the columns in any arbitrary N matrices selected from $\hat{\boldsymbol{H}}_1, \ldots, \hat{\boldsymbol{H}}_{M+N}$ are linearly independent. To generate a basic matrix that satisfies the BLIC, we have Algorithm 3 (see Table III). Based on this algorithm, we have a theorem as follows.

$$\boldsymbol{B}_1=\begin{bmatrix}1&0&0&1&1&0&1\\0&1&0&1&0&1&1\\0&0&1&0&1&1&1\\0&0&0&0&0&0&0\\0&0&0&0&0&0&0\\0&0&0&0&0&0&0\end{bmatrix},\ \boldsymbol{B}_2=\begin{bmatrix}0&0&0&0&0&0&0\\0&0&0&0&0&0&0\\0&0&0&0&0&0&0\\1&0&0&1&1&0&1\\0&1&0&1&0&1&1\\0&0&1&0&1&1&1\end{bmatrix},\ \boldsymbol{B}_3=\begin{bmatrix}1&0&0&1&1&0&1\\0&1&0&1&0&1&1\\0&0&1&0&1&1&1\\1&0&0&1&1&0&1\\0&1&0&1&0&1&1\\0&0&1&0&1&1&1\end{bmatrix},$$

$$\boldsymbol{B}_4=\begin{bmatrix}1&0&0&1&1&0&1\\0&1&0&1&0&1&1\\0&0&1&0&1&1&1\\0&0&1&0&1&1&1\\1&0&1&1&0&1&0\\0&1&0&1&0&1&1\end{bmatrix},\ \boldsymbol{B}_5=\begin{bmatrix}1&0&0&1&1&0&1\\0&1&0&1&0&1&1\\0&0&1&0&1&1&1\\0&1&0&1&0&1&1\\0&1&1&1&1&0&0\\1&0&1&1&0&1&0\end{bmatrix},\ \boldsymbol{B}_6=\begin{bmatrix}1&0&0&1&1&0&1\\0&1&0&1&0&1&1\\0&0&1&0&1&1&1\\1&0&1&1&0&1&0\\1&1&1&0&0&0&1\\0&1&1&1&1&0&0\end{bmatrix},$$

$$\boldsymbol{B}_7=\begin{bmatrix}1&0&0&1&1&0&1\\0&1&0&1&0&1&1\\0&0&1&0&1&1&1\\0&1&1&1&1&0&0\\1&1&0&0&1&1&0\\1&1&1&0&0&0&1\end{bmatrix},\ \boldsymbol{B}_8=\begin{bmatrix}1&0&0&1&1&0&1\\0&1&0&1&0&1&1\\0&0&1&0&1&1&1\\1&1&1&0&0&0&1\\1&0&0&1&1&0&1\\1&1&0&0&1&1&0\end{bmatrix},\ \boldsymbol{B}_9=\begin{bmatrix}1&0&0&1&1&0&1\\0&1&0&1&0&1&1\\0&0&1&0&1&1&1\\1&1&0&0&1&1&0\\0&0&1&0&1&1&1\\1&0&0&1&1&0&1\end{bmatrix}.$$

Fig. 2. Nine binary matrices generated by using Algorithm 2.

TABLE III
DESCRIPTION OF ALGORITHM 3 THAT IS USED TO GENERATE BASIC MATRICES FOR M-SOURCE, N-RELAY NETWORKS ($N>2$)

Algorithm 3 :

Input: Positive integers υ and N ($N>2$).
Output: Binary matrices $\hat{\boldsymbol{H}}_q$, $q=1,2,\cdots,N+2^{\upsilon}+1$.
Steps:

1: Choose a positive integer υ ($\upsilon>2$) such that $f(x)=x^{\upsilon}+x^{\upsilon-1}+1$ is a primitive polynomial over GF(2).
2: Choose $N\upsilon$ linearly independent binary columns of length $N\upsilon$, e.g., columns from the $N\upsilon\times N\upsilon$ identity matrix.
3: Split the $N\upsilon$ columns into N matrices $\boldsymbol{G}_1,\boldsymbol{G}_2,\cdots,\boldsymbol{G}_N$, with υ columns in each matrix.
4: Given n, $n=2,\cdots,N$, generate $2^{\upsilon}-1$ matrices $\boldsymbol{G}_{n,0},\cdots,\boldsymbol{G}_{n,2^{\upsilon}-2}$ based on the columns in $\boldsymbol{G}_n$ according to **Algorithm** 1. Set $\boldsymbol{G}_{n,0}=\boldsymbol{G}_n$.
5: Generate $2^{\upsilon}-1$ matrices by

$$\boldsymbol{G}_{N+1}=\boldsymbol{G}_1\oplus\sum_{n=2}^{N}\oplus\boldsymbol{G}_{n,0},$$

$$\boldsymbol{G}_{N+2}=\boldsymbol{G}_1\oplus\sum_{n=2}^{N}\oplus\boldsymbol{G}_{n,(n-1)\mod(2^{\upsilon}-1)},$$

$$\vdots$$

$$\boldsymbol{G}_{N+k}=\boldsymbol{G}_1\oplus\sum_{n=2}^{N}\oplus\boldsymbol{G}_{n,(k-1)(n-1)\mod(2^{\upsilon}-1)},$$

$$\vdots$$

$$\boldsymbol{G}_{N+2^{\upsilon}-1}=\boldsymbol{G}_1\oplus\sum_{n=2}^{N}\oplus\boldsymbol{G}_{n,(2^{\upsilon}-2)(n-1)\mod(2^{\upsilon}-1)}.$$

6: Randomly choose j ($j=2,\cdots,\upsilon$) columns of the υ columns in $\boldsymbol{G}_k$, $k=1,\cdots,N+2^{\upsilon}+1$, and combine them with a XOR operation to generate a new column. There are a total $\sum_{j=2}^{\upsilon}C_{\upsilon}^{j}=2^{\upsilon}-\upsilon-1$ new columns based on the matrix $\boldsymbol{G}_k$. Together with the original υ columns in $\boldsymbol{G}_k$, we obtain a new matrix $\boldsymbol{B}_k$ composed of these $2^{\upsilon}-1$ columns.
7: Obtain $N+2^{\upsilon}+1$ matrices $\boldsymbol{B}_k$, $k=1,\cdots,N+2^{\upsilon}+1$.
8: Within each matrix $\boldsymbol{B}_k$, randomly choose υ linearly independent columns to construct $\hat{\boldsymbol{H}}_k$.

Theorem 3: If we construct a basic matrix $\hat{\boldsymbol{H}}=[\hat{\boldsymbol{H}}_1,\ldots,\hat{\boldsymbol{H}}_{M+N}]$ by Algorithm 3, then all the columns in any arbitrary N matrices selected from $\hat{\boldsymbol{H}}_1,\ldots,\hat{\boldsymbol{H}}_{M+N}$ are linearly independent, and the basic matrix $\hat{\boldsymbol{H}}$ satisfies the BLIC.

Proof: See Appendix F.

We can see that the BFNC scheme proposed by Algorithm 3 can support an $M-N-1$ relaying network with $M\le 2^{\upsilon}-1$ to achieve $(N+1)$-order diversity.

By using Algorithms 2 and 3, we can construct the codes that satisfy the LIC. In fact, the LIC is equivalent to the design criterion of maximum distance separable (MDS) channel codes. However, the conventional MDS codes may not be suitable for BP decoding. Different from the conventional constructions of MDS codes, our design of BFNC using Algorithms 2 and 3 does not generate a specific code but provides a group of linearly independent vectors. As we will show in the next section, from the vectors generated by these algorithms, we can search a network code with both the optimal degree distribution and MDS property. Therefore, when the code length is large, our BFNC schemes have a good coding gain under BP decoding while achieving full diversity. This is one of the advantages of the BFNC schemes compared with the GFNC schemes.

V. MODIFIED BELIEF PROPAGATION DESIGN FOR LARGE BLOCK LENGTHS

For large block lengths, low encoding complexity is a major advantage of the BFNC schemes proposed in Section IV. From a decoding point of view, the BP decoding can be applied to the proposed BFNC schemes to achieve full diversity, which has low complexity compared to ML decoding for large block lengths. As mentioned in Section III, the parity check matrix $\boldsymbol{H}'$ is used to achieve full diversity with a BP decoder. However, the matrix $\boldsymbol{H}'$ generated (through $\boldsymbol{H}$) by Gaussian elimination could be a dense matrix, which is not suitable for BP iterative decoding since it may lead to a poor coding gain. In the following, we provide an in-depth study on constructing a MBP for the proposed BFNC schemes to obtain full diversity and good coding gain.

The MBP consists of two concatenated BP decoders, as shown in Fig. 3. The first BP decoder receives the channel log-likelihood ratio (LLR) of the transmitted signals and uses $\boldsymbol{H}'$ as the parity check matrix to achieve full diversity. According to the concept of RC-LDPC codes [22], $\boldsymbol{H}'$ can be seen as an

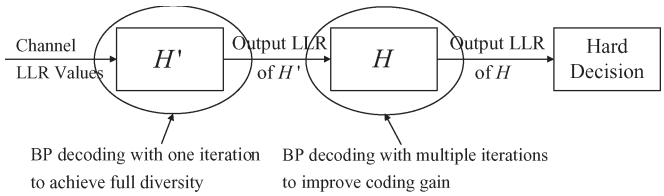


Fig. 3. Concatenated BP decoders composed by $\boldsymbol{H}'$ and $\boldsymbol{H}$. The parity check matrix $\boldsymbol{H}'$ is used first to achieve full diversity. Then, the output LLR values are used as the input of $\boldsymbol{H}$ to achieve more coding gain.

RC-LDPC code. Thus, all variable nodes can achieve full diversity gain after one iteration of the first BP decoder. The output LLR values are then used as the input of the second BP decoder. In this second BP decoder, the parity check matrix $\boldsymbol{H}$ with a low density is used for multiple-iteration processing so as to improve the error performance. Therefore, in our MBP, since $\boldsymbol{H}'$ already guarantees full diversity gain, we need to design $\boldsymbol{H}$ as a good LDPC code to improve the error performance through multiple iterations under BP decoding.

Recall that by the proposed algorithms in Section IV, we can generate a series of binary matrices $\boldsymbol{B}_k$. Based on these matrices, we randomly generate $\hat{\boldsymbol{H}}$ that satisfies the LIC. The parity check matrix $\boldsymbol{H}$ can be obtained by expanding its basic matrix $\hat{\boldsymbol{H}}$. We can view $\boldsymbol{H}$ as a protograph LDPC code [30], and its basic matrix $\hat{\boldsymbol{H}}$ as the protograph. In this protograph LDPC code, the basic matrix $\hat{\boldsymbol{H}}$ serves as a blueprint for constructing $\boldsymbol{H}$ of arbitrary size whose performance can be predicted by analyzing the protograph (i.e., the basic matrix). Using the optimization method in [30], we can obtain a desired $\boldsymbol{H}$ (with a higher threshold) by searching the corresponding basic matrix $\hat{\boldsymbol{H}}$ among all combinations of binary matrices $\boldsymbol{B}_k$.

VI. Numerical Results

A. BLER and BER Performance With Small Block Lengths

First, we consider two $M-2-1$ networks, i.e., the 2-2-1 relaying network and the 3-2-1 relaying network, to illustrate the code design based on the proposed method in this paper. Small block lengths and perfect source-to-relay channels are assumed for each network. The frame lengths in the two networks are the same, i.e., $l=3$, and the block lengths in the two networks are 12 and 15. For these small block lengths, ML decoding can be applied to the destination. In networks with BFNC scheme, we use the basic matrices as the parity check matrix $\boldsymbol{H}$. The basic matrices are obtained by Algorithm 2.

Let us take the 2-2-1 relaying network, for example. In the first step, we obtain the matrix sets $\boldsymbol{B}_k$, $k=1,\ldots,9$, according to Algorithm 2 for the 2-2-1 relaying networks. In the second step, we search the submatrices $\hat{\boldsymbol{H}}_1$, $\hat{\boldsymbol{H}}_2$, $\hat{\boldsymbol{H}}_3$, and $\hat{\boldsymbol{H}}_4$ from all the $\boldsymbol{B}_k$ by utilizing density evolution. The matrices $\hat{\boldsymbol{H}}_1$, $\hat{\boldsymbol{H}}_2$, $\hat{\boldsymbol{H}}_3$, and $\hat{\boldsymbol{H}}_4$ construct the basic matrix $\hat{\boldsymbol{H}}_{2-2-1}$ (see Fig. 4). The matrices $\hat{\boldsymbol{H}}_1$, $\hat{\boldsymbol{H}}_2$, $\hat{\boldsymbol{H}}_3$, and $\hat{\boldsymbol{H}}_4$ are applied to the frames transmitted by s_1, s_2, r_1, and r_2, respectively. Following a similar method, we obtain the basic matrix for the 3-2-1 relaying network. We compare the proposed BFNC scheme with the GFNC [19] scheme and the CFNC scheme [17] in the two networks. All the parameters of the GFNC scheme and the

$$\hat{H}_{2-2-1}=\begin{array}{c}\begin{array}{cccccccccccc}\hat{H}_1 & & & \hat{H}_2 & & & \hat{H}_3 & & & \hat{H}_4 & & \end{array}\\ \left[\begin{array}{ccc|ccc|ccc|ccc} 0&0&1&1&1&0&1&1&1&0&0&0\\ 1&0&1&0&1&0&1&0&1&0&0&0\\ 0&1&1&1&1&1&0&1&1&0&0&0\\ 1&1&1&1&0&0&0&0&0&1&1&1\\ 0&0&1&1&1&1&0&0&0&1&0&1\\ 1&0&0&1&1&0&0&0&0&0&1&1 \end{array}\right]\end{array}$$

$$\hat{H}_{3-3-1}=\begin{array}{c}\begin{array}{cccccccccccccccccc}\hat{H}_1 & & & \hat{H}_2 & & & \hat{H}_3 & & & \hat{H}_4 & & & \hat{H}_5 & & & \hat{H}_6 & & \end{array}\\ \left[\begin{array}{ccc|ccc|ccc|ccc|ccc|ccc} 1&0&0&1&1&0&0&0&1&1&1&1&0&0&0&0&0&0\\ 0&1&0&0&1&1&1&0&0&1&0&1&0&0&0&0&0&0\\ 0&0&1&0&0&1&0&1&1&0&1&1&0&0&0&0&0&0\\ 1&0&0&0&0&1&1&0&1&0&0&0&1&1&1&0&0&0\\ 0&1&0&1&1&1&0&1&1&0&0&0&1&0&1&0&0&0\\ 0&0&1&0&1&1&0&0&1&0&0&0&0&1&1&0&0&0\\ 1&0&0&0&1&1&1&1&0&0&0&0&0&0&0&1&1&1\\ 0&1&0&0&1&0&0&0&1&0&0&0&0&0&0&1&0&1\\ 0&0&1&1&1&1&1&0&0&0&0&0&0&0&0&0&1&1 \end{array}\right]\end{array}$$

Fig. 4. Basic matrices $\hat{\boldsymbol{H}}_{2-2-1}$ and $\hat{\boldsymbol{H}}_{3-3-1}$ designed for the 2-2-1 relaying network and the 3-3-1 relaying network, respectively. In the 2-2-1 relaying network, the submatrices $\hat{\boldsymbol{H}}_1$, $\hat{\boldsymbol{H}}_2$, $\hat{\boldsymbol{H}}_3$, and $\hat{\boldsymbol{H}}_4$ are applied to the frames transmitted by s_1, s_2, r_1, and r_2, respectively. In the 3-3-1 relaying network, the submatrices $\hat{\boldsymbol{H}}_1$, $\hat{\boldsymbol{H}}_2$, $\hat{\boldsymbol{H}}_3$, $\hat{\boldsymbol{H}}_4$, $\hat{\boldsymbol{H}}_5$, and $\hat{\boldsymbol{H}}_6$ are applied to the frames transmitted by s_1, s_2, s_3, r_1, r_2, and r_3, respectively.

CFNC scheme are optimally chosen according to [19] and [17], respectively.

For the GFNC scheme, we choose the Galois field with eight elements, i.e., GF(8). At the relay encoder, we map a frame (three bits) into a Galois field element (or GF symbol) and then generate the network-coded GF symbols by the GFNC scheme according to [19]. Specifically, in the 2-2-1 relaying network, at the first relay, we choose the two GF(8) elements 1 and 1 to combine the GF symbols from the two sources. The NC process at the first relay can be expressed as $\boldsymbol{x}_{r_1}^{GFNC}=1\cdot\boldsymbol{x}_{s_1}\boxplus 1\cdot\boldsymbol{x}_{s_2}$, where $\boxplus$ represents the plus operation in the corresponding Galois field. At the second relay, we use the two GF(8) elements 1 and 2 to combine the frames from the two sources. The NC process at the second relay can be expressed as $\boldsymbol{x}_{r_1}^{GFNC}=1\cdot\boldsymbol{x}_{s_1}\boxplus 2\cdot\boldsymbol{x}_{s_2}$. Similarly, in the 3-2-1 relaying network, at the first relay the GF(8) vector [1 1 1], and at the second relay, the GF(8) vector [1 2 3] is utilized to combine the GF symbols from the three sources. For the CFNC scheme, space–time codes are applied to the relays. The transmitted frame by the relay r_n, i.e., $\boldsymbol{x}_{r_n}^{CFNC}$, is written as $\boldsymbol{x}_{r_n}^{CFNC}=1/\sqrt{M}\sum_{m=1}^{M}\theta_{n,m}\boldsymbol{x}_{s_m}$, where the coefficients $\theta_{n,1},\ldots,\theta_{n,M}$ are from a space–time coding matrix. According to [17], we have $\theta_{1,1}=1$, $\theta_{1,2}=e^{j(3\pi/4)}$, $\theta_{2,1}=1$, and $\theta_{2,2}=e^{j(7\pi/4)}$ for the 2-2-1 relaying network. In the 3-2-1 relaying network, we have $\theta_{1,1}=1$, $\theta_{1,2}=e^{j(5\pi/9)}$, $\theta_{1,3}=e^{j(10\pi/9)}$, $\theta_{2,1}=1$, $\theta_{2,2}=e^{j(11\pi/9)}$, and $\theta_{2,3}=e^{j(22\pi/9)}$.

Fig. 5 shows the error probabilities of the 2-2-1 relaying network with BFNC, CFNC, and GFNC with block length $b_{block}=12$. Note that the solid black lines represent the block error probabilities (BLEP) and the dashed black lines represent the bit error probabilities (BEPs). Fig. 6 shows the error probabilities of the 3-2-1 relaying network under the three NC schemes. We can see from Figs. 5 and 6 that all three NC

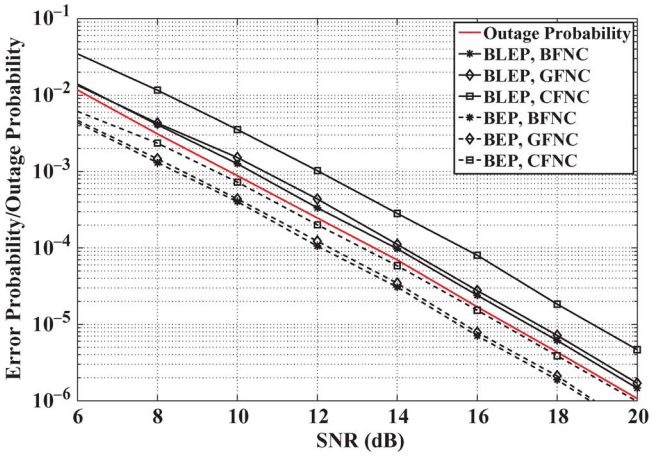


Fig. 5. Error probabilities of the 2-2-1 relaying network under the three NC schemes. The block length is 12, and the source-to-relay channels are perfect. An ML decoder is used at the destination. Solid lines represent the block error probabilities, and dashed lines represent the BEPs.

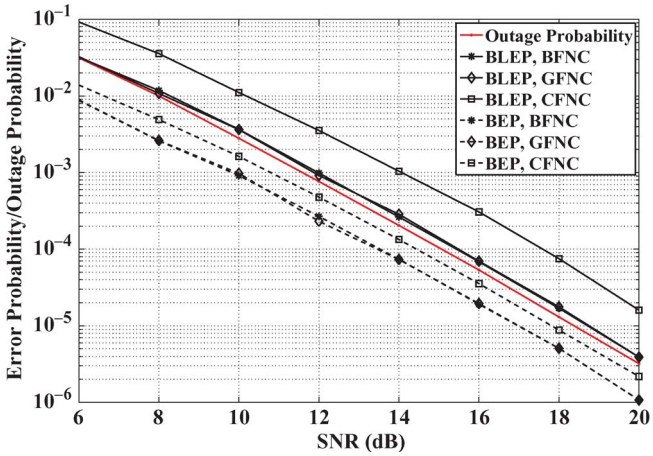


Fig. 6. Error probabilities for the 3-2-1 relaying network under the three NC schemes. The block length is 15, and the source-to-relay channels are perfect. An ML decoder is used at the destination. Solid lines represent block error probabilities, and dashed lines represent the BEPs.

schemes can achieve full diversity (third-order diversity) in the 2-2-1 and 3-2-1 relaying networks. The BFNC scheme and the GFNC scheme outperform the CFNC scheme in terms of error performance. We also find that the BFNC scheme is slightly better than the GFNC scheme.

Second, we consider the general $M - N - 1$ relaying networks. Without loss of generality, we consider a 3-3-1 relaying network. Small block length and perfect source-to-relay channels are assumed for the network. The frame length is $l = 3$, and the block length is 18. The ML decoding is applied to the destination. According to Algorithm 3, we obtain the basic matrix $\hat{\boldsymbol{H}}_{3-3-1}$ for the 3-3-1 relaying network, as shown in Fig. 4. Fig. 7 shows the error probabilities of the 3-3-1 relaying network with BFNC, GFNC, and CFNC. Note that the solid black lines represent the block error probabilities (BLEP), and the dashed black lines represent the BEPs. We can see from Fig. 7 that the BFNC scheme can achieve full diversity (fourth-order diversity) in the 3-3-1 relaying network. We also found that the error performance of the BFNC scheme is similar to

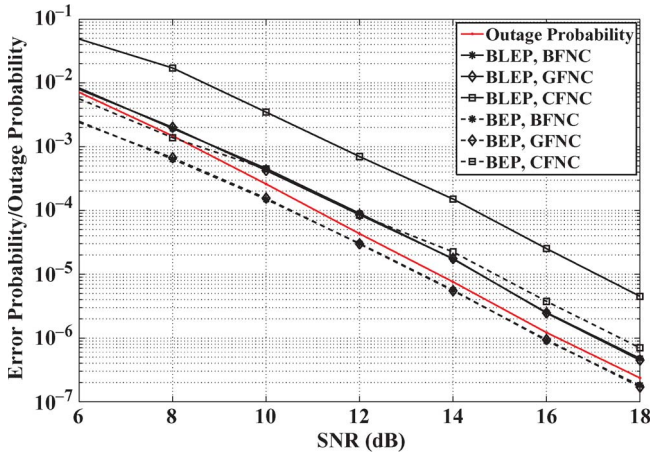


Fig. 7. Error probability for the 3-3-1 relaying network under the three NC schemes. The block length is 18, and the source-to-relay channels are perfect. An ML decoder is used at the destination. Solid lines represent the BLEP, and dashed lines represent the BEP.

that of the GFNC scheme. In addition, both BFNC and GFNC schemes outperform the CFNC scheme.

B. BLER Performance Under Imperfect Source-to-Relay Channels

We investigate the BFNC, CFNC, and GFNC schemes in the 2-2-1 network with imperfect source-to-relay channels. As shown in [17] and [19], the full-diversity achievability of both GFNC and CFNC is not changed by imperfect source-relay channels. Utilizing a method similar to that of [17] and [19], we show that the BFNC scheme can also achieve full diversity in imperfect source-to-relay channels. Fig. 8 shows the BLEP and BEP curves of the three NC schemes under imperfect source-to-relay channels. We assume that the fading coefficients at all the source-to-relay channels are Rayleigh distributed with zero mean and unit variance. The additive channel noises at the relays are Gaussian distributed with variance σ^2. We use the BLEP curve of the BFNC scheme under error-free source-to-relay channels as a benchmark. We can see that all three schemes can still achieve full diversity under the imperfect source-to-relay channels. Both BFNC and GFNC schemes have around 1.1-dB performance gain relative to the CFNC scheme.

C. BLER Performance With Large Block Lengths

We consider the 2-2-1 relaying network with large block lengths. We assume that the frame length is $l = 300$ and the source-to-relay channels are perfect. Thus, the block length in the network is $l_{\text{block}} = 1200$. For the BFNC scheme with a block length of 1200, we generate the 600×1200 parity check matrix $\boldsymbol{H}$ using our proposed algorithms. The MBP is applied to the destination. Fig. 9 shows the BLEP of the 2-2-1 relaying network with the BFNC scheme. In Fig. 9, we compare the proposed MBP with the conventional BP decoder, which is based on the parity check matrix $\boldsymbol{H}$. We consider 0, 4, and 100 iterations of the parity check matrix $\boldsymbol{H}$ in the MBP decoder and 100 iterations in the BP decoder. Note that in Fig. 9, the "0 iteration" of the parity check matrix $\boldsymbol{H}$ means that we do

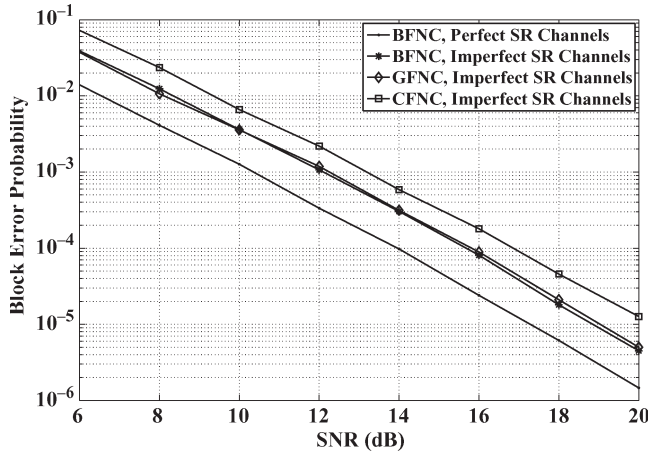


Fig. 8. Block error probabilities for the 2-2-1 relaying network under the three NC schemes. Imperfect source-to-relay channels are considered. Block length is 12. An ML decoder is used at the destination.

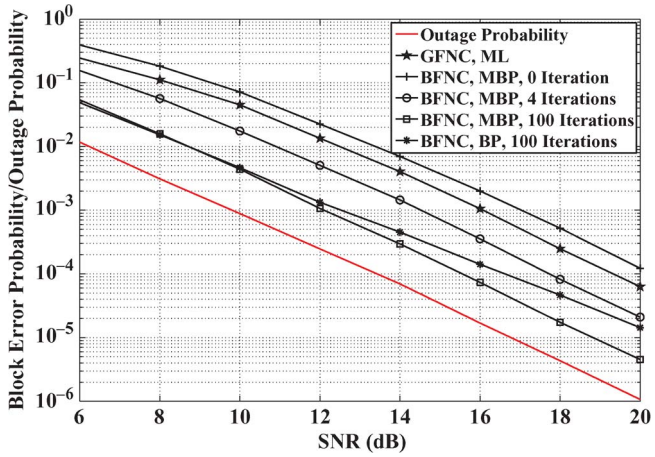


Fig. 9. Block error probabilities for the 2-2-1 relaying network under the BFNC and GFNC schemes. The block length is 1200. The proposed BP decoder is used at the destination in the BFNC scheme, and the ML decoder is used in the GFNC scheme.

not use $\boldsymbol{H}$ for decoding. In the case of "0 iteration," we make the decision based on the output LLR of the diversity achieving decoder $\boldsymbol{H}'$. From Fig. 9, we can see that due to the RC-LDPC matrix $\boldsymbol{H}'$ and its corresponding diversity evolution, the MBP decoder with "0 iteration," "4 iterations," and "100 iterations" can all achieve the full diversity gain (third-order diversity). The case of "100 iterations" in $\boldsymbol{H}$ facilitates higher coding gain. We can also see from Fig. 9 that without using $\boldsymbol{H}'$ and diversity evolution, the iterative BP decoder based on $\boldsymbol{H}$ with "100 iterations" cannot achieve full diversity.

In Fig. 9 the GFNC code is chosen over GF(8), i.e., we choose two GF(8) elements 1 and 1 to combine the GF symbols from the two sources at the first relay, and we use two GF(8) elements 1 and 2 to combine the frames from the two sources at the second relay. In Fig. 9, the GFNC scheme can achieve full diversity. However, when compared to the BFNC scheme with 'MBP' and "100 iterations," the coding gain of GFNC is 4-dB worse.

We can see that in CFNC and GFNC, the complexity of ML decoding increases according to 2^{Ml}, whereas in BFNC, as we can use BP decoding, the decoding complexity possesses a linear complexity in terms of frame length [30]. Therefore, our BFNC is efficient relative to CFNC and GFNC for large frame lengths.

VII. Conclusion

In this paper, we have studied a binary field NC design over a multiple-source multiple-relay wireless network over slow-fading channels. We have discussed a diversity-achieving criterion for the BFNC schemes with either an ML decoder or a BP decoder at the destination. Based on this criterion, we then proposed algorithms that construct low-complexity encoders based on frame-wise cyclic-shifting matrices, which are then used to generate our new BFNC schemes. Numerical results demonstrate that our BFNC schemes outperform previous complex field and GFNC schemes in that our BFNC schemes can achieve full diversity gain and high coding gain for arbitrary block lengths with low encoding/decoding complexity.

Appendix A
Proof of Lemma 1

We define the received signal vector as $\boldsymbol{y}=[\boldsymbol{y}_{11}^T,\ldots,\boldsymbol{y}_{1M}^T,\boldsymbol{y}_{21}^T,\ldots,\boldsymbol{y}_{2N}^T]^T$, and we have $\boldsymbol{y}=\boldsymbol{X}_{\text{diag}}\boldsymbol{h}+\boldsymbol{v}$, where $\boldsymbol{X}_{\text{diag}}=\text{diag}(\boldsymbol{x}_{s_1},\ldots,\boldsymbol{x}_{s_M},\boldsymbol{x}_{r_1},\ldots,\boldsymbol{x}_{r_N})$, $\boldsymbol{h}=[\hbar_1,\ldots,\hbar_M,h_1,\ldots,h_N]^T$ and $\boldsymbol{v}=[\boldsymbol{v}_{11}^T,\ldots,\boldsymbol{v}_{1M}^T,\boldsymbol{v}_{21}^T,\ldots,\boldsymbol{v}_{2N}^T]^T$. We define that $\hat{\boldsymbol{X}}_{\text{diag}}=\text{diag}(\hat{\boldsymbol{x}}_{s_1},\ldots,\hat{\boldsymbol{x}}_{s_M},\hat{\boldsymbol{x}}_{r_1},\ldots,\hat{\boldsymbol{x}}_{r_N})$. In addition, we denote the decoding power normalized error matrix as $\boldsymbol{U}=1/\sqrt{P}(\boldsymbol{X}_{\text{diag}}-\hat{\boldsymbol{X}}_{\text{diag}})$ and the autocorrelation matrix of $\boldsymbol{v}$ as $\boldsymbol{F}_v$. According to [27], we obtain the PEP as

$$P(\boldsymbol{x}\to\hat{\boldsymbol{x}})=\frac{1}{\pi}\int_0^{\frac{\pi}{2}}\mathbb{E}_{\boldsymbol{h}}\left\{\exp\left(-P\frac{\boldsymbol{h}^H\boldsymbol{U}^H\boldsymbol{F}_{\boldsymbol{v}}^{-1}\boldsymbol{U}\boldsymbol{h}}{8\sin^2\theta}\right)\right\}d\theta \tag{7}$$

where $\mathbb{E}_{\boldsymbol{h}}$ is the mathematical expectation over $\boldsymbol{h}$. Note that for a random column vector $\boldsymbol{z}$ with zero mean and autocorrelation matrix $\boldsymbol{F}_{\boldsymbol{z}}$, and a Hermitian matrix $\boldsymbol{H}$, we have $\mathbb{E}\{\exp(-\boldsymbol{z}^H\boldsymbol{H}\boldsymbol{z})\}=1/\det(\boldsymbol{I}+\boldsymbol{F}_{\boldsymbol{z}}\boldsymbol{H})$. In (7), we take the expectation with respect to $\boldsymbol{h}$ and notice that $\boldsymbol{U}^H\boldsymbol{U}$ is an $(M+N)\times(M+N)$ Hermitian matrix. Then, we obtain

$$P(\boldsymbol{x}\to\hat{\boldsymbol{x}})=\frac{1}{\pi}\int_0^{\frac{\pi}{2}}\frac{1}{\det\left(\boldsymbol{I}+\left(\frac{\rho}{8\sin^2\theta}\right)\boldsymbol{U}^H\boldsymbol{U}\right)}\,d\theta. \tag{8}$$

When ρ is large, we can replace $\det(\boldsymbol{I}+(\rho/8\sin^2\theta)\boldsymbol{U}^H\boldsymbol{U})$ with $(\rho/8\sin^2\theta)^{(M+N)}\det((1/\rho)\boldsymbol{I}+\boldsymbol{U}^H\boldsymbol{U})$. Since

$$\det\left(\frac{1}{\rho}\boldsymbol{I}+\boldsymbol{U}^H\boldsymbol{U}\right)=\prod_{m=1}^{M}\left(\frac{1}{\rho}+\|\boldsymbol{u}_{s_m}\|^2\right)\cdot\prod_{n=1}^{N}\left(\frac{1}{\rho}+\|\boldsymbol{u}_{r_n}\|^2\right)$$

$$\int_0^{\frac{\pi}{2}}\sin^{2(M+N)}\theta\,d\theta=\frac{1\cdot3\cdot5\cdots(2M+2N-1)}{2\cdot4\cdot6\cdots(2M+2N)}\cdot\frac{\pi}{2} \tag{9}$$

we obtain (2) and complete the proof. ■

APPENDIX B
PROOF OF THEOREM 1

We randomly choose two binary blocks generated by $\boldsymbol{H}$, i.e., $\boldsymbol{b} = [\boldsymbol{b}_{s_1}^T, \ldots, \boldsymbol{b}_{s_M}^T, \boldsymbol{b}_{r_1}^T, \ldots, \boldsymbol{b}_{r_N}^T]^T$ and $\hat{\boldsymbol{b}} = [\hat{\boldsymbol{b}}_{s_1}^T, \ldots, \hat{\boldsymbol{b}}_{s_M}^T, \hat{\boldsymbol{b}}_{r_1}^T, \ldots, \hat{\boldsymbol{b}}_{r_N}^T]^T$. The modulated blocks are $\boldsymbol{x} = (-1)^{\boldsymbol{b}}$, $\hat{\boldsymbol{x}} = (-1)^{\hat{\boldsymbol{b}}}$. In addition, we have $\sum_{k=1}^{M+N} \oplus \boldsymbol{H}_k(\boldsymbol{b}_{s_k} \oplus \hat{\boldsymbol{b}}_{s_k}) = \boldsymbol{o}$.

To find the minimum distance between the two blocks, we generate $M+N$ bit vectors as $\boldsymbol{b}_{s_1} \oplus \hat{\boldsymbol{b}}_{s_1}, \ldots, \boldsymbol{b}_{s_M} \oplus \hat{\boldsymbol{b}}_{s_M}$, $\boldsymbol{b}_{r_1} \oplus \hat{\boldsymbol{b}}_{r_1}, \ldots, \boldsymbol{b}_{r_N} \oplus \hat{\boldsymbol{b}}_{r_N}$. If $\boldsymbol{H}$ is so designed that the columns in any N of matrices $\boldsymbol{H}_1, \ldots, \boldsymbol{H}_{M+N}$ are linearly independent, then among these $M+N$ bit vectors, there are at least $N+1$ non-zero vectors. Correspondingly, there are at least $N+1$ non-zero vectors in $\boldsymbol{x}_{s_1} - \hat{\boldsymbol{x}}_{s_1}, \ldots, \boldsymbol{x}_{s_M} - \hat{\boldsymbol{x}}_{s_M}, \boldsymbol{x}_{r_1} - \hat{\boldsymbol{x}}_{r_1}, \ldots, \boldsymbol{x}_{r_N} - \hat{\boldsymbol{x}}_{r_N}$. This means that the minimum frame-wise Hamming distance of arbitrary two blocks $\boldsymbol{x}$ and $\hat{\boldsymbol{x}}$ is $N+1$. According to Lemma 1, the diversity gain of the $M-N-1$ relaying network can achieve $(N+1)$-order diversity. This concludes the proof. ■

APPENDIX C
PROOF OF LEMMA 2

In the second step of Algorithm 1, since υ is chosen to make $f(x) = x^{\upsilon} + x^{\upsilon-1} + 1$ primitive over GF(2), we can obtain $2^{\upsilon} - 1$ different nonzero columns $\boldsymbol{b}_k$, $k = 1, \ldots, 2^{\upsilon} - 1$. From the third step of Algorithm 1, we can see that each matrix $\boldsymbol{G}_{2,k}$ possesses at least one column that does not appear in all the other matrices. Therefore, $\boldsymbol{G}_{2,k}$ are $2^{\upsilon} - 1$ unique matrices.

It is clear that the columns in $\boldsymbol{G}_{2,1}$ are linearly independent. Then, we focus on $\boldsymbol{G}_{2,2}$. Let

$$\gamma_1 \boldsymbol{b}_2 \oplus \gamma_2 \boldsymbol{b}_3 \oplus \cdots \oplus \gamma_{\upsilon-1} \boldsymbol{b}_{\upsilon} \oplus \gamma_{\upsilon} \boldsymbol{b}_{\upsilon+1} = \boldsymbol{o} \tag{10}$$

where $\gamma_1, \ldots, \gamma_{\upsilon} \in \{0, 1\}$ are binary coefficients. Since $\boldsymbol{b}_{\upsilon+1} = \boldsymbol{b}_1 \oplus \boldsymbol{b}_2$, we can rewrite (10) as

$$\gamma_{\upsilon} \boldsymbol{b}_1 \oplus (\gamma_1 \oplus \gamma_{\upsilon}) \boldsymbol{b}_2 \oplus \gamma_2 \boldsymbol{b}_3 \oplus \cdots \oplus \gamma_{\upsilon-1} \boldsymbol{b}_{\upsilon} = \boldsymbol{o}. \tag{11}$$

Since $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{\upsilon}$ are linearly independent, we have $\gamma_1 \oplus \gamma_{\upsilon} = \gamma_2 = \cdots = \gamma_{\upsilon} = 0$, which indicates that $\gamma_1 = \gamma_2 = \cdots = \gamma_{\upsilon} = 0$. Therefore, the columns in $\boldsymbol{G}_{2,2}$, i.e., $\boldsymbol{b}_2, \ldots, \boldsymbol{b}_{\upsilon+1}$, are linearly independent. In $\boldsymbol{G}_{2,3}$, we have the columns $\boldsymbol{b}_3, \ldots, \boldsymbol{b}_{\upsilon+2}$, where $\boldsymbol{b}_{\upsilon+2} = \boldsymbol{b}_2 \oplus \boldsymbol{b}_3$. Based on the result that $\boldsymbol{b}_2, \ldots, \boldsymbol{b}_{\upsilon+1}$ are linearly independent, we can obtain that $\boldsymbol{b}_3, \ldots, \boldsymbol{b}_{\upsilon+2}$ are linearly independent. By doing this recursively (i.e., we prove the columns in $\boldsymbol{G}_{2,k}$ to be linearly independent based on the result that the columns in $\boldsymbol{G}_{2,k-1}$ are linearly independent), we can finally prove that the columns in each $\boldsymbol{G}_{2,k}$, $k = 1, \ldots, 2^{\upsilon} - 1$, are linearly independent. ■

APPENDIX D
PROOF OF LEMMA 3

We define a series of matrices $\boldsymbol{D}_1, \boldsymbol{D}_2, \ldots, \boldsymbol{D}_{2^{\upsilon}-1}$ as follows: $\boldsymbol{D}_1 \triangleq [\boldsymbol{b}_1, \boldsymbol{b}_2, \ldots, \boldsymbol{b}_{2^{\upsilon}-1}]^T$, $\boldsymbol{D}_2 \triangleq [\boldsymbol{b}_2, \boldsymbol{b}_3, \ldots, \boldsymbol{b}_{2^{\upsilon}-1}, \boldsymbol{b}_1]^T, \ldots, \boldsymbol{D}_{2^{\upsilon}-1} \triangleq [\boldsymbol{b}_{2^{\upsilon}-1}, \boldsymbol{b}_1, \boldsymbol{b}_2, \ldots, \boldsymbol{b}_{2^{\upsilon}-2}]^T$. We have $\boldsymbol{D}_2 = \boldsymbol{I}^{2^{\upsilon}-1}(1)\boldsymbol{D}_1$, $\boldsymbol{D}_3 = \boldsymbol{I}^{2^{\upsilon}-1}(2)\boldsymbol{D}_1, \ldots, \boldsymbol{D}_{2^{\upsilon}-1} = \boldsymbol{I}^{2^{\upsilon}-1}(2^{\upsilon}-2)\boldsymbol{D}_1$, where the matrix $\boldsymbol{I}^{2^{\upsilon}-1}(\alpha)$, as defined in Section IV, is a circulant permutation matrix that circularly shifts the $(2^{\upsilon} - 1) \times (2^{\upsilon} - 1)$ identity matrix to the right by α times, $\alpha \in \{1, \ldots, 2^{\upsilon} - 2\}$. To prove Lemma 3, we will first prove that among the matrices set $\boldsymbol{D}_1, \boldsymbol{D}_2, \ldots, \boldsymbol{D}_{2^{\upsilon}-1}$, the addition of any two different matrices over GF(2) is another matrix in this set, i.e., for arbitrary i and j ($i, j \in \{1, \ldots, 2^{\upsilon} - 1\}$ and $i \neq j$), there exists $q \in \{1, \ldots, 2^{\upsilon} - 1\}$ such that $\boldsymbol{D}_i \oplus \boldsymbol{D}_j = \boldsymbol{D}_q$.

Note that $\boldsymbol{b}_1, \boldsymbol{b}_2, \ldots, \boldsymbol{b}_{\upsilon}$ are υ linearly independent binary vectors with length equal to or larger than υ. Binary vectors $\boldsymbol{b}_{\upsilon+1}, \boldsymbol{b}_{\upsilon+2}, \ldots, \boldsymbol{b}_{2^{\upsilon}-1}$ are linear combinations [over GF(2)] of $\boldsymbol{b}_1, \boldsymbol{b}_2, \ldots, \boldsymbol{b}_{\upsilon}$ based on the primitive GF(2) polynomial $f(x) = x^{\upsilon} + x^{\upsilon-1} + 1$, as described in Algorithm 1. Then, we denote a length $(2^{\upsilon} - 1)$ binary vector $\boldsymbol{f}$ as the coefficients vector of $f(x)$, i.e.,

$$\boldsymbol{f} = [\underbrace{0 \cdots 0}_{2^{\upsilon}-\upsilon-2\text{times}} \; 1 \; 1 \; \underbrace{0 \cdots 0}_{\upsilon-2\text{times}} \; 1]. \tag{12}$$

Equation (12) implies that the three "1"s in $\boldsymbol{f}$ are located at the first position, the υth position, and the $(\upsilon+1)$th position from the right to the left, respectively. By circularly shifting the coefficients vector $\boldsymbol{f}$ to the left by β times, $\beta = 1, \ldots, 2^{\upsilon} - \upsilon - 2$, we obtain $2^{\upsilon} - \upsilon - 2$ new binary vectors of length $(2^{\upsilon} - 1)$, which are denoted as $\boldsymbol{f}_{\beta}$. Note that for each β, the vector $\boldsymbol{f}_{\beta}$ is the coefficients vector of the polynomial $x^{\beta} f(x)$. According to Algorithm 1, $\boldsymbol{b}_1 \oplus \boldsymbol{b}_2 \oplus \boldsymbol{b}_{\upsilon+1} = \boldsymbol{o}$, $\boldsymbol{b}_2 \oplus \boldsymbol{b}_3 \oplus \boldsymbol{b}_{\upsilon+2} = \boldsymbol{o}, \ldots, \boldsymbol{b}_{2^{\upsilon}-\upsilon-2} \oplus \boldsymbol{b}_{2^{\upsilon}-\upsilon-1} \oplus \boldsymbol{b}_{2^{\upsilon}-2} = \boldsymbol{o}$, and $\boldsymbol{b}_{2^{\upsilon}-\upsilon-1} \oplus \boldsymbol{b}_{2^{\upsilon}-\upsilon} \oplus \boldsymbol{b}_{2^{\upsilon}-1} = \boldsymbol{o}$, where $\boldsymbol{o}$ is a zero vector. Correspondingly

$$\begin{aligned}
\boldsymbol{f}_{2^{\upsilon}-\upsilon-2}\boldsymbol{D}_1 &= [1 \; 1 \; \underbrace{0 \cdots 0}_{\upsilon-2\text{times}} \; 1 \; \underbrace{0 \cdots 0}_{2^{\upsilon}-\upsilon-2\text{times}}]\boldsymbol{D}_1 = \boldsymbol{o}^T \\
\boldsymbol{f}_{2^{\upsilon}-\upsilon-3}\boldsymbol{D}_1 &= [0 \; 1 \; 1 \; \underbrace{0 \cdots 0}_{\upsilon-2\text{times}} \; 1 \; \underbrace{0 \cdots 0}_{2^{\upsilon}-\upsilon-3\text{times}}]\boldsymbol{D}_1 = \boldsymbol{o}^T \\
&\vdots \qquad\qquad \vdots \qquad\qquad \vdots \\
\boldsymbol{f}_1\boldsymbol{D}_1 &= [\underbrace{0 \cdots 0}_{2^{\upsilon}-\upsilon-3\text{times}} \; 1 \; 1 \; \underbrace{0 \cdots 0}_{\upsilon-2\text{times}} \; 1 \; 0]\boldsymbol{D}_1 = \boldsymbol{o}^T \\
\boldsymbol{f}\boldsymbol{D}_1 &= [\underbrace{0 \cdots 0}_{2^{\upsilon}-\upsilon-2\text{times}} \; 1 \; 1 \; \underbrace{0 \cdots 0}_{\upsilon-2\text{times}} \; 1]\boldsymbol{D}_1 = \boldsymbol{o}^T.
\end{aligned} \tag{13}$$

Furthermore, since binary vectors $\boldsymbol{b}_{\upsilon+1}, \boldsymbol{b}_{\upsilon+2}, \ldots, \boldsymbol{b}_{2^{\upsilon}-1}$ are linear combinations of $\boldsymbol{b}_1, \boldsymbol{b}_2, \ldots, \boldsymbol{b}_{\upsilon}$ over GF(2), the matrix $\boldsymbol{D}_1$ can be written as $\boldsymbol{D}_1 = \boldsymbol{Q}[\boldsymbol{b}_1, \boldsymbol{b}_2, \ldots, \boldsymbol{b}_{\upsilon}]^T$, where $\boldsymbol{Q}$ is a $(2^{\upsilon} - 1) \times \upsilon$ binary coefficient matrix. Since $f(x)$ is primitive, each $\boldsymbol{b}_k$, $k \in \{1, \ldots, 2^{\upsilon} - 1\}$, is a unique combination of $\boldsymbol{b}_1$, $\boldsymbol{b}_2, \ldots, \boldsymbol{b}_{\upsilon}$. Therefore, all the rows in $\boldsymbol{Q}$ are nonzero and are different from each other. Note that for a given vector length υ, we can obtain a total of $2^{\upsilon} - 1$ unique nonzero binary vectors, which constitute all the $2^{\upsilon} - 1$ rows of $\boldsymbol{Q}$. Therefore,

addition operations among all these $2^\upsilon - 1$ rows of $\boldsymbol{Q}$ are closed over GF(2), i.e., the addition of arbitrary two rows over GF(2) is another row in $\boldsymbol{Q}$. Hence, for arbitrary i and j ($i, j \in \{1, \ldots, 2^\upsilon - 1\}$ and $i \neq j$), there exists $q \in \{1, \ldots, 2^\upsilon - 1\}$ such that $\boldsymbol{b}_i \oplus \boldsymbol{b}_j = \boldsymbol{b}_q$, i.e., $\boldsymbol{b}_i \oplus \boldsymbol{b}_j \oplus \boldsymbol{b}_q = \boldsymbol{o}$.

Based on the triplet $\{i, j, q\}$, we construct a polynomial $g(x) = x^{2^\upsilon - i - 1} + x^{2^\upsilon - j - 1} + x^{2^\upsilon - q - 1}$ over GF(2). We denote a length $(2^\upsilon - 1)$ binary nonzero vector $\boldsymbol{g}$ as the coefficient vector of $g(x)$. The three "1"s in $\boldsymbol{g}$ are located at the ith position, the jth position, and the qth position from the left to the right, respectively. Since $\boldsymbol{b}_i \oplus \boldsymbol{b}_j \oplus \boldsymbol{b}_q = \boldsymbol{o}$, we have $\boldsymbol{g}\boldsymbol{D}_1 = \boldsymbol{o}^T$. Note that $g(x)$ can be written as $g(x) = h(x) \cdot f(x) + d(x)$, where $h(x)$ and $d(x)$ are over GF(2), and

$$\begin{aligned} h(x) =& 0x^{2^\upsilon - 2} + \cdots + 0x^{2^\upsilon - \upsilon - 1} + h_{2^\upsilon - \upsilon - 2}x^{2^\upsilon - \upsilon - 2} \\ &+ h_{2^\upsilon - \upsilon - 3}x^{2^\upsilon - \upsilon - 3} + \cdots + h_0 \\ d(x) =& 0x^{2^\upsilon - 2} + \cdots + 0x^{\upsilon} + d_{\upsilon - 1}x^{\upsilon - 1} \\ &+ d_{\upsilon - 2}x^{\upsilon - 2} + \cdots + d_0. \end{aligned} \tag{14}$$

In (14), the coefficients $h_0, \ldots, h_{2^\upsilon - \upsilon - 2} \in \{0, 1\}$ and $d_0, \ldots, d_{\upsilon - 1} \in \{0, 1\}$. We denote the length $(2^\upsilon - 1)$ coefficient vectors of polynomials $h(x)f(x)$ and $d(x)$ as $\boldsymbol{u}$ and $\boldsymbol{d}$, respectively. Vectors $\boldsymbol{u}$, $\boldsymbol{d}$, and $\boldsymbol{g}$ satisfy $\boldsymbol{u} \oplus \boldsymbol{d} = \boldsymbol{g}$. From (14), the polynomial $h(x)f(x)$ can be written as

$$\begin{aligned} h(x)f(x) =& h_{2^\upsilon - \upsilon - 2}x^{2^\upsilon - \upsilon - 2}f(x) \\ &+ h_{2^\upsilon - \upsilon - 3}x^{2^\upsilon - \upsilon - 3}f(x) + \cdots + h_0 f(x). \end{aligned} \tag{15}$$

Since $\boldsymbol{f}_\beta$, $\beta = 1, \ldots, 2^\upsilon - \upsilon - 2$, are the coefficients vectors of $x^\beta f(x)$, the vector $\boldsymbol{u}$ is a linear combination of $\boldsymbol{f}$ and $\boldsymbol{f}_\beta$ over GF(2), i.e.,

$$\boldsymbol{u} = h_{2^\upsilon - \upsilon - 2}\boldsymbol{f}_{2^\upsilon - \upsilon - 2} \oplus h_{2^\upsilon - \upsilon - 3}\boldsymbol{f}_{2^\upsilon - \upsilon - 3} \oplus \cdots \oplus h_0 \boldsymbol{f}. \tag{16}$$

From (13), we have $\boldsymbol{u}\boldsymbol{D}_1 = \boldsymbol{o}^T$, which, combined with the fact that $\boldsymbol{g}\boldsymbol{D}_1 = \boldsymbol{o}^T$, implies that

$$\boldsymbol{d}\boldsymbol{D}_1 = d_{\upsilon - 1}\boldsymbol{b}^T_{2^\upsilon - \upsilon} + d_{\upsilon - 2}\boldsymbol{b}^T_{2^\upsilon - \upsilon + 1} + \cdots + d_0\boldsymbol{b}^T_{2^\upsilon - 1} = \boldsymbol{o}^T. \tag{17}$$

Since $\boldsymbol{G}_{2, 2^\upsilon - \upsilon} = [\boldsymbol{b}_{2^\upsilon - \upsilon}, \ldots, \boldsymbol{b}_{2^\upsilon - 1}]$, according to Lemma 2, the υ vectors $\boldsymbol{b}_{2^\upsilon - \upsilon}, \ldots, \boldsymbol{b}_{2^\upsilon - 1}$ are linearly independent. Then, we have $d_{\upsilon - 1} = \cdots = d_0 = 0$. Therefore, $d(x) = 0$ and $g(x) = h(x)f(x)$, i.e., $g(x)$ is divisible by $f(x)$.

By circularly shifting $\boldsymbol{g}$ to the right by p times, $p = 1, \ldots, 2^\upsilon - 2$, we obtain another $2^\upsilon - 2$ vector, which can be expressed as $\boldsymbol{g}\boldsymbol{I}^{2^\upsilon - 1}(p)$. For each p, we use the vector $\boldsymbol{g}\boldsymbol{I}^{2^\upsilon - 1}(p)$ as the coefficients vector to define the polynomial $g_p(x)$. The polynomial $g_p(x)$ can be written as $g_p(x) = x^{-p}g(x) \bmod (x^{2^\upsilon - 1} + 1)$. Since both $g(x)$ and $(x^{2^\upsilon - 1} + 1)$ are divisible by the primitive polynomial $f(x)$, each $g_p(x)$ is divisible by $f(x)$, which means that $\boldsymbol{g}\boldsymbol{I}^{2^\upsilon - 1}(p)\boldsymbol{D}_1 = \boldsymbol{o}^T$ for each p. Therefore, we have

$$\begin{bmatrix} \boldsymbol{g} \\ \boldsymbol{g}\boldsymbol{I}^{2^\upsilon - 1}(1) \\ \vdots \\ \boldsymbol{g}\boldsymbol{I}^{2^\upsilon - 1}(2^\upsilon - 2) \end{bmatrix} \boldsymbol{D}_1 = \boldsymbol{O} \tag{18}$$

where $\boldsymbol{O}$ is a zero matrix. Note that in (18), the first row is $\boldsymbol{g}$, in which the three "1"s are located at the ith position, the jth position, and the qth position from the left to the right. The $(p+1)$th row in the matrix $p = 1, \ldots, 2^\upsilon - 2$ is obtained by circularly shifting the first row to the right by p times. Therefore

$$\begin{aligned} &\begin{bmatrix} \boldsymbol{g} \\ \boldsymbol{g}\boldsymbol{I}^{2^\upsilon - 1}(1) \\ \vdots \\ \boldsymbol{g}\boldsymbol{I}^{2^\upsilon - 1}(2^\upsilon - 2) \end{bmatrix} \\ &\quad = \boldsymbol{I}^{2^\upsilon - 1}(i-1) \oplus \boldsymbol{I}^{2^\upsilon - 1}(j-1) \oplus \boldsymbol{I}^{2^\upsilon - 1}(q-1). \end{aligned} \tag{19}$$

We have

$$\left(\boldsymbol{I}^{2^\upsilon - 1}(i-1) \oplus \boldsymbol{I}^{2^\upsilon - 1}(j-1) \oplus \boldsymbol{I}^{2^\upsilon - 1}(q-1)\right)\boldsymbol{D}_1 = \boldsymbol{O}. \tag{20}$$

Since $\boldsymbol{D}_i = \boldsymbol{I}^{2^\upsilon - 1}(i-1)\boldsymbol{D}_1$, $\boldsymbol{D}_j = \boldsymbol{I}^{2^\upsilon - 1}(j-1)\boldsymbol{D}_1$, and $\boldsymbol{D}_q = \boldsymbol{I}^{2^\upsilon - 1}(q-1)\boldsymbol{D}_1$, (20) becomes $\boldsymbol{D}_i \oplus \boldsymbol{D}_j \oplus \boldsymbol{D}_q = \boldsymbol{O}$, i.e., $\boldsymbol{D}_i \oplus \boldsymbol{D}_j = \boldsymbol{D}_q$. Note that $\boldsymbol{G}_{2,k}$ is formed by using the first υ columns of the matrix $\boldsymbol{D}^T_k$. Thus, we have $\boldsymbol{G}_{2,i} \oplus \boldsymbol{G}_{2,j} = \boldsymbol{G}_{2,q}$. Therefore, we complete the proof. ■

APPENDIX E
PROOF OF THEOREM 2

Suppose that $\boldsymbol{G}_k = [\boldsymbol{b}_{k,1}, \ldots, \boldsymbol{b}_{k,\upsilon}]_{2\upsilon \times \upsilon}$, $k = 1, \ldots, 2^\upsilon + 1$, where $\boldsymbol{b}_{k,j}$, $j = 1, \ldots, \upsilon$, is the jth column in $\boldsymbol{G}_k$. First, according to Lemma 2, we can prove that all the columns in each $\boldsymbol{G}_k$ are linearly independent. In addition, all the columns in the matrix $[\boldsymbol{G}_1, \boldsymbol{G}_i]$, $i = 3, \ldots, 2^\upsilon + 1$, are linearly independent. All the columns in the matrix $[\boldsymbol{G}_2, \boldsymbol{G}_i], i = 3, \ldots, 2^\upsilon + 1$ are linearly independent. In the next step, we need to prove that all the columns in the matrix $[\boldsymbol{G}_{i_1}, \boldsymbol{G}_{i_2}]$ are linearly independent, where $i_1 \neq i_2$, and $i_1, i_2 = 3, \ldots, 2^\upsilon + 1$.

For all the columns from the matrix $[\boldsymbol{G}_{i_1}, \boldsymbol{G}_{i_2}]$, we let

$$\sum_{j=1}^{\upsilon} \oplus \beta_j \boldsymbol{b}_{i_1,j} \oplus \sum_{j'=1}^{\upsilon} \oplus \beta_{\upsilon + j'} \boldsymbol{b}_{i_2,j'} = \boldsymbol{o} \tag{21}$$

where the coefficients $\beta_1, \ldots, \beta_{2\upsilon}$ are binary numbers. Then, we have

$$\sum_{j=1}^{\upsilon} \oplus (\beta_j \oplus \beta_{\upsilon + j})\boldsymbol{b}_{1,j} \oplus F(\boldsymbol{b}_{2,1}, \ldots, \boldsymbol{b}_{2,\upsilon}) = \boldsymbol{o} \tag{22}$$

where the function $F(\boldsymbol{b}_{2,1}, \ldots, \boldsymbol{b}_{2,2^\upsilon} - 1)$ is a linear combination of vectors $\boldsymbol{b}_{2,1}, \ldots, \boldsymbol{b}_{2,2^\upsilon - 1}$ in the binary field. Note that $F(\boldsymbol{b}_{2,1}, \ldots, \boldsymbol{b}_{2,2^\upsilon} - 1)$ is linearly independent on

$\boldsymbol{b}_{1,1},\ldots,\boldsymbol{b}_{1,\upsilon}$. In addition, $\boldsymbol{b}_{1,1},\ldots,\boldsymbol{b}_{1,\upsilon}$ are linearly independent. We have $\beta_1=\beta_{\upsilon+1},\ldots,\beta_\upsilon=\beta_{2\upsilon}$. Therefore, (21) can be rewritten as

$$\sum_{j=1}^{\upsilon}\oplus\beta_j(\boldsymbol{b}_{i_1,j}\oplus\boldsymbol{b}_{i_2,j})=\boldsymbol{o}. \tag{23}$$

According to Lemma 3, the matrix $[\boldsymbol{b}_{i_1,1}\oplus\boldsymbol{b}_{i_2,1},\ldots,\boldsymbol{b}_{i_1,\upsilon}\oplus\boldsymbol{b}_{i_2,\upsilon}]$ is one of $\boldsymbol{G}_{2,1},\ldots,\boldsymbol{G}_{2,\upsilon}$, which means that $\boldsymbol{b}_{i_1,1}\oplus\boldsymbol{b}_{i_2,1},\ldots,\boldsymbol{b}_{i_1,\upsilon}\oplus\boldsymbol{b}_{i_2,\upsilon}$ are linearly independent. Therefore, we have $\beta_1=\cdots=\beta_\upsilon=0$, and the binary columns in (21) are linearly independent.

We can then conclude that all the columns in the matrix $[\boldsymbol{G}_{i_1},\boldsymbol{G}_{i_2}]$ are linearly independent, where $i_1\neq i_2$, and $i_1,i_2=1,\ldots,2^\upsilon+1$. This is equivalent to stating that the matrix $[\boldsymbol{G}_{i_1},\boldsymbol{G}_{i_2}]$ is full rank. Note that the columns in the matrix $\hat{\boldsymbol{H}}_k$ are linearly independent and from the linear combinations of the columns of $\boldsymbol{G}_k$. Therefore, all the columns in the matrix $[\hat{\boldsymbol{H}}_{i_1},\hat{\boldsymbol{H}}_{i_2}]$ are linearly independent, $i_1\neq i_2$, and $i_1,i_2=1,\ldots,2^\upsilon+1$. This concludes the proof. ■

APPENDIX F
PROOF OF THEOREM 3

Suppose that $\boldsymbol{G}_k=[\boldsymbol{b}_{k,1},\ldots,\boldsymbol{b}_{k,\upsilon}]_{2\upsilon\times\upsilon}$, $k=1,\ldots,N+2^\upsilon-1$, where $\boldsymbol{b}_{k,j},j=1,\ldots,\upsilon$, is the jth column in $\boldsymbol{G}_k$. According to Algorithm 3, $\boldsymbol{G}_k$, $k=1,\ldots,N$, are constructed so that all the columns in the matrix $[\boldsymbol{G}_1,\boldsymbol{G}_2,\ldots,\boldsymbol{G}_N]$ are linearly independent. Then, we focus on the matrices $\boldsymbol{G}_{N+1}$, $\boldsymbol{G}_{N+2},\ldots,\boldsymbol{G}_{N+2^\upsilon+1}$. Randomly choosing N matrices from $\boldsymbol{G}_{N+1},\boldsymbol{G}_{N+2},\ldots,\boldsymbol{G}_{N+2^\upsilon+1}$, we obtain $\boldsymbol{G}_{i_1},\boldsymbol{G}_{i_2},\ldots,\boldsymbol{G}_{i_N}$, where $i_1\neq i_2\neq\cdots\neq i_N$, and $i_1,i_2,\ldots,i_N\in\{N+1,\ldots,N+2^\upsilon-1\}$. We will prove that all the columns in the matrix $[\boldsymbol{G}_{i_1},\boldsymbol{G}_{i_2},\ldots,\boldsymbol{G}_{i_N}]$ are linearly independent. For $n=1,\ldots,N$, we have

$$\boldsymbol{G}_{i_n}=\boldsymbol{G}_1\oplus\sum_{\eta=2}^{N}\oplus\boldsymbol{G}_{\eta,(i_n-N-1)(\eta-1)\mod(2^\upsilon-1)}. \tag{24}$$

According to Lemma 2, all the columns in the matrix $\boldsymbol{G}_{\eta,(i_n-N-1)(\eta-1)\mathrm{mod}(2^\upsilon-1)}$ in (24) are linearly independent, and then all the columns in the matrix $\boldsymbol{G}_{i_n}$, i.e., $\boldsymbol{b}_{i_n,1},\ldots,\boldsymbol{b}_{i_n,\upsilon}$, are linearly independent. Then, we consider the vectors $\boldsymbol{b}_{i_1,j},\ldots,\boldsymbol{b}_{i_N,j}$, where $j\in\{1,\ldots,\upsilon\}$. Based on (24), we have

$$\boldsymbol{b}_{i_n,j}=\boldsymbol{b}_{1,j}\oplus\sum_{\eta=2}^{N}\oplus\boldsymbol{b}_{\eta,(i_n-N-1)(\eta-1)\mod(2^\upsilon-1),j} \tag{25}$$

where $\boldsymbol{b}_{\eta,(i_n-N-1)(\eta-1)\mathrm{mod}(2^\upsilon-1),j}$ is the jth column of $\boldsymbol{G}_{\eta,(i_n-N1)(\eta-1)\mathrm{mod}(2^\upsilon-1)}$.

To prove that all the columns in the matrix $[\boldsymbol{G}_{i_1},\boldsymbol{G}_{i_2},\ldots,\boldsymbol{G}_{i_N}]$ are linearly independent, we let

$$\sum_{j=1}^{\upsilon}\oplus\sum_{n=1}^{N}\oplus\beta_{n,j}\boldsymbol{b}_{i_n,j}=\boldsymbol{o} \tag{26}$$

where each coefficient $\beta_{n,j}$ is a binary number. Plugging (25) into (26), we have

$$\begin{aligned}&\left(\sum_{j=1}^{\upsilon}\oplus\sum_{n=1}^{N}\oplus\beta_{n,j}\boldsymbol{b}_{1,j}\right)\\&\oplus\left(\sum_{j=1}^{\upsilon}\oplus\sum_{n=1}^{N}\oplus\beta_{n,j}\boldsymbol{b}_{2,(i_n-N-1)\mathrm{mod}(2^\upsilon-1),j}\right)\oplus\cdots\\&\oplus\left(\sum_{j=1}^{\upsilon}\oplus\sum_{n=1}^{N}\oplus\beta_{n,j}\boldsymbol{b}_{N,(i_n-N-1)(N-1)\mathrm{mod}(2^\upsilon-1),j}\right)=\boldsymbol{o}.\end{aligned} \tag{27}$$

Since the columns in the matrix $[\boldsymbol{G}_1,\boldsymbol{G}_2,\ldots,\boldsymbol{G}_N]$ are linearly independent, $\boldsymbol{b}_{1,j},\boldsymbol{b}_{2,j},\ldots,\boldsymbol{b}_{N,j}$ are linearly independent. Therefore, from (27), we can obtain the equation group as

$$\begin{aligned}&\sum_{j=1}^{\upsilon}\oplus\sum_{n=1}^{N}\oplus\beta_{n,j}\boldsymbol{b}_{1,j}=\boldsymbol{o}\\&\sum_{j=1}^{\upsilon}\oplus\sum_{n=1}^{N}\oplus\beta_{n,j}\boldsymbol{b}_{2,(i_n-N-1)\mod(2^\upsilon-1),j}=\boldsymbol{o}\\&\quad\vdots\\&\sum_{j=1}^{\upsilon}\oplus\sum_{n=1}^{N}\oplus\beta_{n,j}\boldsymbol{b}_{N,(i_n-N-1)(N-1)\mod(2^\upsilon-1),j}=\boldsymbol{o}.\end{aligned} \tag{28}$$

From (28), we obtain $\sum_{n=1}^{N}\oplus\beta_{n,j}=0$, and thus, we have $\beta_{N,j}=\sum_{n=1}^{N-1}\oplus\beta_{n,j}$. We rewrite (26) as

$$\sum_{n=1}^{N-1}\oplus\sum_{j=1}^{\upsilon}\oplus\beta_{n,j}(\boldsymbol{b}_{i_n,j}\oplus\boldsymbol{b}_{i_N,j})=\boldsymbol{o}. \tag{29}$$

Furthermore, the subscript of the vectors $(i_1-N-1)(n-1)\mathrm{mod}(2^\upsilon-1)$ indicates that the vectors

$$\begin{aligned}&\left[\boldsymbol{b}_{1,j}^T,\boldsymbol{b}_{2,(i_1-N-1)\mathrm{mod}(2^\upsilon-1),j}^T,\cdots\right.\\&\qquad\left.\boldsymbol{b}_{N,(i_1-N-1)(N-1)\mathrm{mod}(2^\upsilon-1),j}^T\right]^T\\&\qquad\vdots\\&\left[\boldsymbol{b}_{1,j}^T,\boldsymbol{b}_{2,(i_N-N-1)\mathrm{mod}(2^\upsilon-1),j}^T\cdots\right.\\&\qquad\left.\boldsymbol{b}_{N,(i_N-N-1)(N-1)\mathrm{mod}(2^\upsilon-1),j}^T\right]^T\end{aligned}$$

are linearly independent. Therefore, we get according to Lemma 3 that columns $\boldsymbol{b}_{i_n,j}\oplus\boldsymbol{b}_{i_N,j}$ are linearly independent for $n=1,\ldots,N-1$ and $j=1,\ldots,\upsilon$. Therefore, we have the coefficient $\beta_{n,j}=0$ for $n=1,\ldots,N-1$ and $j=1,\ldots,\upsilon$. Consequently, we have $\beta_{N,j}=0$ for all j. Based on this, we immediately get that all the columns in the matrix $[\boldsymbol{G}_{i_1},\boldsymbol{G}_{i_2},\ldots,\boldsymbol{G}_{i_N}]$ are linearly independent, where $i_1\neq i_2\neq\cdots\neq i_N$, and $i_1,i_2,\ldots,i_N\in\{N+1,\ldots,N+2^\upsilon-1\}$. In addition, note that all the columns in the matrix $[\boldsymbol{G}_1,\boldsymbol{G}_2,\ldots,\boldsymbol{G}_N]$ are linearly independent. Then, we obtain that all the columns in $[\boldsymbol{G}_{j_1},\boldsymbol{G}_{j_2},\ldots,\boldsymbol{G}_{j_N}]$ are linearly independent, where $j_1\neq j_2\neq\cdots\neq j_N$, and $j_1,j_2,\ldots,j_N\in\{1,\ldots,N+2^\upsilon-1\}$. This concludes the proof. ■

REFERENCES

[1] T. M. Cover and A. A. E. Gamal, "Capacity theorems for the relay channel," *IEEE Trans. Inf. Theory*, vol. IT-25, no. 5, pp. 572–584, Sep. 1979.

[2] A. Sendonaris, E. Erkip, and B. Azhang, "User cooperation diversity—Part I: System description," *IEEE Trans. Commun.*, vol. 51, no. 11, pp. 1927–1938, Nov. 2003.

[3] J. Kim, D. S. Michalopoulos, and R. Schober, "Diversity analysis of multi-user multi-relay networks," *IEEE Trans. Wireless Commun.*, vol. 10, no. 7, pp. 2380–2389, Jul. 2011.

[4] Z. Wang, W. Chen, F. Gao, and J. Li, "Capacity performance of efficient relay beamformings for dual-hop MIMO multi-relay networks with imperfect R-D CSI at relays," *IEEE Trans. Veh. Technol.*, vol. 60, no. 6, pp. 2608–2619, Jul. 2011.

[5] T. Yang, J. Yuan, and W. Zhang, "Recovering cooperative multiplexing gain in wireless relay networks," *IEEE Trans. Commun.*, vol. 58, no. 12, pp. 3538–3549, Dec. 2010.

[6] M. H. Azmi, J. Yuan, G. Lechner, and L. K. Rasmussen, "Design of multi-edge type bilayer-expurgated LDPC codes for decode-and-forward in relay channels," *IEEE Trans. Commun.*, vol. 59, no. 11, pp. 2993–3006, Nov. 2011.

[7] N. Yang, M. Elkashlan, and J. Yuan, "Outage probability of multiuser relay networks in Nakagami-m fading channels," *IEEE Trans. Veh. Technol.*, vol. 59, no. 5, pp. 2120–2132, Jun. 2010.

[8] M. Peng, Y. Liu, D. Wei, W. Wang, and H.-H. Chen, "Hierarchical cooperative relay based heterogeneous networks," *IEEE Wireless Commun.*, vol. 18, no. 3, pp. 48–56, Jun. 2011.

[9] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung, "Network information flow," *IEEE Trans. Inf. Theory*, vol. 46, no. 4, pp. 1204–1216, Jul. 2000.

[10] S. Zhang and S.-C. Liew, "Channel coding and decoding in a relay system operated with physical-layer network coding," *IEEE J. Sel. Areas Commun.*, vol. 27, no. 5, pp. 788–796, Jun. 2009.

[11] Y. Li, R. Louie, and B. Vucetic, "Relay selection with network coding in two-way relay channels," *IEEE Trans. Veh. Technol.*, vol. 59, no. 9, pp. 4489–4499, Nov. 2010.

[12] H. V. Nguyen, S. X. Ng, J. L. Rebelatto, Y. Li, and L. Hanzo, "Near-capacity network coding for cooperative multi-user communications," in *Proc. IEEE VTC*, Sep. 2011, pp. 1–5.

[13] W. Li, J. Li, and P. Fan, "Network coding for two-way relaying networks over Rayleigh fading channels," *IEEE Trans. Veh. Technol.*, vol. 59, no. 9, pp. 4476–4488, Nov. 2010.

[14] K. Chi, X. Jiang, and S. Horiguchi, "Joint design of network coding and transmission rate selection for multihop wireless networks," *IEEE Trans. Veh. Technol.*, vol. 59, no. 5, pp. 2435–2444, Jun. 2010.

[15] J. Park, S.-L. Kim, and J. Choi, "Hierarchically modulated network coding for asymmetric two-way relay systems," *IEEE Trans. Veh. Technol.*, vol. 59, no. 5, pp. 2179–2184, Jun. 2010.

[16] Z. Ding and K. K. Leung, "On the combination of cooperative diversity and network coding for wireless uplink transmissions," *IEEE Trans. Veh. Technol.*, vol. 60, no. 4, pp. 1590–1601, May 2011.

[17] T. Wang and G. B. Giannakis, "Complex field network coding for multi-user cooperative communications," *IEEE J. Sel. Areas Commun.*, vol. 26, no. 3, pp. 561–571, Apr. 2008.

[18] M. Peng, H. Liu, W. Wang, and H.-H. Chen, "Cooperative network coding with MIMO transmission in wireless decode-and-forward relay networks," *IEEE Trans. Veh. Technol.*, vol. 59, no. 7, pp. 3577–3588, Sep. 2010.

[19] M. Xiao and M. Skoglund, "Design of network codes for multiple-user multiple-relay wireless networks," in *Proc. IEEE ISIT*, Jun. 2009, pp. 2562–2566.

[20] M. Xiao and M. Skoglund, "Multiple-user cooperative communications based on linear network coding," *IEEE Trans. Commun.*, vol. 58, no. 12, pp. 3345–3351, Dec. 2010.

[21] J. L. Rebelatto, B. F. Uchoa-Filho, Y. Li, and B. Vucetic, "Generalized distributed network coding based on nonbinary linear block codes for multi-user cooperative communications," in *Proc. IEEE ISIT*, Jun. 2010, pp. 943–947.

[22] D. Duyck, D. Capirone, J. J. Boutros, and M. Moeneclaey, "Analysis and construction of full-diversity joint network-LDPC codes for cooperative communications," *EURASIP J. Wireless Commun. Netw.*, vol. 2010, p. 9, Jan. 2010.

[23] K. Pang, Z. Lin, Y. Li, and B. Vucetic, "Design of distributed network-channel codes for wireless sensor networks," in *Proc. IEEE ICC*, Jun. 2011, pp. 1–5.

[24] J. Li, J. Yuan, R. Malaney, M. H. Azmi, and M. Xiao, "Network coded LDPC code design for a multi-source relaying system," *IEEE Trans. Wireless Commun.*, vol. 10, no. 5, pp. 1538–1551, May 2011.

[25] Y. Li, G. Song, and L. Wang, "Design of joint network-low density parity check codes based on the EXIT charts," *IEEE Commun. Lett.*, vol. 13, no. 8, pp. 600–602, Aug. 2009.

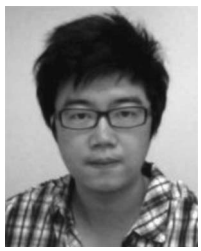
[26] H. V. Nguyen, S. X. Ng, and L. Hanzo, "Performance bounds of network coding aided cooperative multiuser systems," *IEEE Signal Process. Lett.*, vol. 18, no. 7, pp. 435–438, Jul. 2011.

[27] J. Li and W. Chen, "Joint power allocation and precoding for network coding based cooperative multicast systems," *IEEE Signal Process. Lett.*, vol. 15, pp. 817–820, 2008.

[28] N.-H. Quttineh, "Computational complexity of finite field multiplication," Ph.D. dissertation, Linköping Univ., Linköping, Sweden, 2003.

[29] S.-N. Hong, S. Kim, D.-J. Shin, and I. Lee, "Quasi-cyclic low-density parity check codes for space-time bit-interleaved coded modulation," *IEEE Commun. Lett.*, vol. 12, no. 10, pp. 767–769, Oct. 2008.

[30] J. Thorpe, "Low density parity check (LDPC) codes constructed from protographs," Jet Propulsion Lab., Pasadena, CA, INP Progr. Rep. 42-154, Aug. 2003.



Jun Li (M'09) received the Ph.D. degree in electronic engineering from Shanghai Jiaotong University, Shanghai, China, in 2009.

From January 2009 to June 2009, he was a Research Scientist with the Department of Research and Innovation, Alcatel Lucent Shanghai Bell. Since June 2009, he has been a Research Fellow with the School of Electrical Engineering and Telecommunications, University of New South Wales, Sydney, Australia. His research interests include network information theory, channel coding theory, wireless network coding, and cooperative communications.

Dr. Li served as a Technical Program Committee member for the Asia–Pacific Conference on Communications in 2009 and 2010, the IEEE Vehicular Technology Conference in the Spring of 2011, and the International Conference on Communications in 2011.

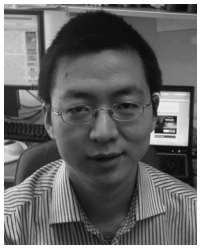
Jinhong Yuan (M'97–SM'11) received the B.E. and Ph.D. degrees in electronics engineering from Beijing Institute of Technology, Beijing, China, in 1991 and 1997, respectively.

From 1997 to 1999, he was a Research Fellow with the School of Electrical Engineering, the University of Sydney, Sydney, Australia. In 2000, he joined the School of Electrical Engineering and Telecommunications, the University of New South Wales, Sydney, where he is currently a Professor for telecommunications. He has published two books, two book chapters, over 190 papers in telecommunications journals and conference proceedings, and 40 industrial reports. He is a co-inventor of one patent on multiple-input–multiple-output systems and two patents on low-density-parity-check codes. His current research interests include error control coding and information theory, communication theory, and wireless communications.

Dr. Yuan serves as the IEEE New South Wales Chair of joint Communications/Signal Processions/Ocean Engineering Chapter. He coauthored three papers that won Best Paper Awards and one Best Poster Award, including a Best Paper Award from the IEEE Wireless Communications and Networking Conference, Cancun, Mexico, in 2011, and a Best Paper Award from the IEEE International Symposium on Wireless Communications Systems, Trondheim, Norway, in 2007.

Robert Malaney (M'03) received the Bachelor of Science degree in physics from the University of Glasgow, Glasgow, U.K., and the Ph.D. degree in physics from the University of St. Andrews, St. Andrews, U.K.

He was a former Principal Research Scientist with CSIRO. He has previously held research positions at the California Institute of Technology, Pasadena; the University of California Berkeley–National Labs, and the University of Toronto, Toronto, ON, Canada. He is currently an Associate Professor with the School of Electrical Engineering and Telecommunications, University of New South Wales, Sydney, Australia. He has over 100 publications.



Ming Xiao (S'02–M'07) received the Bachelor and Master degrees in engineering from the University of Electronic Science and Technology of China, ChengDu, China, in 1997 and 2002, respectively, and the Ph.D. degree from Chalmers University of Technology, Göteborg, Sweden, in November 2007.

He was a Visiting Researcher with the Laboratory for Information and Decision System, Massachusetts Institute of Technology, Cambridge, in 2006 and the Institute of Network Coding, the Chinese University of Hong Kong, Hong Kong, in 2010. From 1997 to 1999, he worked as a Network and Software Engineer with ChinaTelecom. From 2000 to 2002, he also held a position with the SiChuan communications administration. Since November 2007, he has been with the ACCESS Linnaeus Center, School of Electrical Engineering, Royal Institute of Technology, Stockholm, Sweden, where he is currently an Assistant Professor.

Dr. Xiao received the "Chinese Government Award for Outstanding Self-Financed Students Studying Abroad" in March 2007. He received a "Hans Werthen Grant" from the Royal Swedish Academy of Engineering Science (IVA) in March 2006. He received "Ericsson Research Funding" from Ericsson in 2010. He received Best Paper Awards from the International Conference on Wireless Communications and Signal Processing in 2010 and the International Conference on Computer Communication Networks in 2011.

Wen Chen (M'03–SM'11) received the B.S. and M.S. degrees from Wuhan University, Wuhan, China, in 1990 and 1993, respectively, and Ph.D. degree from the University of Electro-Communications, Tokyo, Japan, in 1999.

He was a Researcher with the Japan Society for the Promotion of Sciences from 1999 to 2001. In 2001, he joined the University of Alberta, Edmonton, AB, Canada, starting as a Post-Doctoral Fellow with the Information Research Laboratory and continuing as a Research Associate with the Department of Electrical and Computer Engineering. Since 2006, he has been a Full Professor with the Department of Electronic Engineering, Shanghai Jiaotong University, Shanghai, China, where he is also the Director of the Institute for Signal Processing and Systems. He has published more than 100 papers in IEEE journals and conferences. His interests cover network coding, cooperative communications, cognitive radio, and multiple-input–multiple-output orthogonal frequency-division multiplexing systems.

Dr. Chen received the Ariyama Memorial Research Prize in 1997 and the PIMS Post-Doctoral Fellowship in 2001. He received the honors of "New Century Excellent Scholar in China" in 2006 and the "Pujiang Excellent Scholar in Shanghai" in 2007. He is elected to the Vice General Secretary of the Shanghai Institute of Electronics in 2008. He is in the editorial board of the *International Journal of Wireless Communications and Networking* and serves on the *Journal of Communications*, *Journal of Computers*, *Journal of Networks*, and the EURASIP *Journal on Wireless Communications and Networking* as (lead) Guest Editor. He was the Technical Program Committee Chair for the IEEE International Conference on Circuits and Systems for Communications in 2008, the General Conference Chair for the IEEE International Conference on Computer and Information Science in 2009, and the International Conference on Wireless Communications, Networking, and Information Security in 2010.