Design of network-coding based multi-edge type LDPC codes for multi-source relaying systems

Jun Li, Marwan H. Azmi, Robert. Malaney, and Jinhong Yuan
School of Electrical Engineering and Telecommunications,
University of New South Wales, Sydney, NSW 2052, AUSTRALIA
Email: jun.li@unsw.edu.au, marwan@student.unsw.edu.au, r.malaney@unsw.edu.au, jinhong@ee.unsw.edu.au

Abstract—In this paper we investigate a multi-source LDPC scheme for a Gaussian relay system, where \( M \) sources communicate with the destination under the help of a single relay (\( M - 1 - 1 \) system). Since various distributed LDPC schemes in the cooperative single-source system, e.g. bilayer LDPC and bilayer multi-edge type LDPC (BMET-LDPC) \( 2 \), have been designed to approach the Shannon limit, these schemes can be applied to the \( M - 1 - 1 \) system by the relay serving each source in a round-robin fashion. However, such a direct application is not optimal due to the lack of potential joint processing gain. In this paper, we propose a network coded multi-edge type LDPC (NCMET-LDPC) scheme for the multi-source scenario. Through an EXIT analysis, we conclude that the NCMET-LDPC scheme achieves higher extrinsic mutual information, relative to a separate application of BMET-LDPC to each source. Our new NCMET-LDPC scheme thus achieves a higher threshold relative to existing schemes.

Index Terms—Multi-source LDPC, network coding, network capacity, extrinsic mutual information.

I. INTRODUCTION

Low-density parity-check (LDPC) codes have been shown to approach theoretical capacity limits for single link communication channels \( 1 \). Recently, distributed LDPC for cooperative communications has attracted much attention. The work of \( 2 \) first explored the use of bilayer LDPC codes within the cooperative single source channel (\( 1 - 1 - 1 \) system), where full-duplexing relay is used. Although bilayer LDPC is carefully designed to approach the system capacity \( 3 \), the performance is decreased as the capacity gap between the source-to-relay channel and the source-to-destination channel becomes larger. In \( 5 \), multi-edge type LDPC code has been utilized to address this problem \( 3 \). The works of \( 6 \), \( 7 \), \( 8 \) consider more practical issues in the \( 1 - 1 - 1 \) system, such as the use of Rayleigh fading channels and half-duplexing relays.

However, the above studies on distributed LDPC codes are all limited to the triangle model, which contains only one source. In this paper, we investigate a network coding \( 9 \) based LDPC codes designed for the cooperative uplink system with multi-source and one relay (\( M - 1 - 1 \) system) as shown in Fig. \( 1 \). Based on existing methods of distributed LDPC design in the triangle model, an intuitive thought is that the relay serves the sources in a round-robin fashion, optimizing the distributed LDPC for a single source in each round. Unfortunately, such a direct application is not optimal for the following reasons.

1) The check digits produced by the relay in the \( m \)-th, \( (m = 1, \cdots, M) \), round are only based on the codeword from the source \( s_m \), which is highly correlated with the original check codes already produced by \( s_m \).
2) Network coded check digits enable the joint decoding of all sources’ data and thus each source can obtain more extrinsic mutual information from the other sources.
3) The code profile optimization executed by the relay aims only at approaching the capacity for the single source, which leads to a sub-optimal outcome for a multi-source system in terms of network capacity.

We note that \( 10 \) has studied the network coded LDPC in a multi-source system with fading channels. But the code design in \( 10 \) is not optimal. \( 11 \) also proposes a joint bilayer LDPC scheme in the \( M - 1 - 1 \) system. However, this scheme is constrained by the drawbacks of bilayer LDPC. So it cannot deal with the problem where the capacity gap between source-to-relay channel and source-to-destination is large. Also, \( 11 \) only considers a scenario where all channel capacities in the model are equal. In this paper, we propose a network coding based multi-edge type LDPC, which we refer to as NCMET-LDPC, for the \( M - 1 - 1 \) model which addresses these issues. In our analysis of NCMET-LDPC we utilize extrinsic mutual information (EMI) transfer, rather than density evolution (DE) used in \( 11 \), which aids in understanding the code design issues better. Through the EXIT chart, we see that the network coded parity check digits produced by relay provide more EMI for each source, relative to a separate round-robin scheme.

II. SYSTEM MODEL AND PRELIMINARIES

Consider a Gaussian relay system with \( M \)-sources, 1-relay and 1 destination as shown in Fig. \( 1 \) where sources \( s_1, \cdots, s_M \) transmit information to the destination \( d \) simultaneously with the help of a full-duplex relay \( r \). We assume
all sources are randomly distributed around the relay. Suppose that the $m$-th source, $s_m$, transmits its information in the frequency band $f_m$, and $r$ can receive and transmit at $f_1, \ldots, f_M$. All the frequency bands are assumed to be orthogonal. With these constraints the multi-source system can be viewed as $M$ independent parallel $1 - 1 - 1$ systems. Within each of these channels, a bilayer LDPC scheme [2] can be utilized to approach the $1 - 1 - 1$ system capacity [4].

Let us briefly review the achievable rate and code design in the $1 - 1 - 1$ system [2]. Without loss of generality, we focus firstly on a specific $1 - 1 - 1$ system. This is formed by selecting a specific value of $m$, and constructing a model composed of $s_m, r_m$ and $d$, where $r_m$ is the part of $r$ operating at $f_m$. In Fig. 2, $X_m$ is the signal transmitted by $s_m$, which has the average power $P_m$, and $X_{1m}$ is the signal transmitted by $r_m$, which has the average power $P_{1m}$. The binning scheme is used to achieve the capacity in a Gaussian degraded relay channel [4]. In this binning scheme, $s_m$ divides its total power $P_m$ into a fraction $\alpha P_m$ for the new codeword $\omega_i$, and a fraction $(1 - \alpha)P_m$ for the bin index $\phi_i$ of the previous codeword $\omega_{i-1}$. So in the $i$-th time slot, $X_m$ is the superposition of $\omega_i$ and $\phi_i$, which will be received by both $r_m$ and $d$. Since $r_m$ has successfully decoded $\omega_{i-1}$ in the previous time slot, $\omega_i$ will be successfully decoded at the relay with a rate no more than

$$R^m_+ = \frac{1}{2} \log \left( 1 + \frac{\alpha P_m}{N_{1m}} \right).$$

Meanwhile, $r_m$ is transmitting $X_{1,m}(\phi_i)$ to $d$. Thus, $d$ receives the interferential signal composed by $X_m$ and $X_{1m}$ in the frequency band $f_m$. By successive interference cancellation, $d$ firstly treats $X_m$ as noise so as to extract the bin index $\phi_i$ with a rate no more than

$$R^n_1 = \frac{1}{2} \log \left( 1 + \frac{\sqrt{P_1 m} + \sqrt{(1 - \alpha)P_m}}{\alpha P_m + N_{1m} + N_{2m}} \right).$$

Then combining with $\phi_i$, the decoding of $\omega_{i-1}$ at $d$ will be successful with a rate no more than

$$R^n_m = \frac{1}{2} \log \left( 1 + \frac{\alpha P_m}{N_{1m} + N_{2m}} \right).$$

Combining the above three equations, we get the overall rate for the Gaussian relay channel working at $f_m$ as

$$R_m = \max_{\alpha} \min \{ R^m_+, R^n_1 + R^n_m \}. \quad (4)$$

So to maximize $R_m$, we let $R^m_+ = R^n_1 + R^n_m$ by adapting power allocation $\alpha$.

For the practical code design, we can utilize the bilayer LDPC scheme for each $s_m$ ($m = 1, \ldots, M$) to approach $R^m_+$ and $R^n_m$ [2]. For example, the code structure is divided into the lower and the upper graphs and consists of $k_{1m}$ and $k_{2m}$ check nodes, respectively. Both type of check nodes are connected to the same $n_m$ variable nodes. According to [2], in the design of such codes, one should firstly determine the optimal LDPC code corresponding to the lower graph to achieve rate $R^n_m$, and then search over the whole bilayer graph in order to find a good LDPC code that approaches the rate $R^m_+$. So in the multi-source case, a straightforward technique of practical code design is to perform bilayer LDPC individually to each source. As previously mentioned, such individual processing does not benefit from any joint processing gain at the relay. In the following section, we focus on the network coded multi-edge type LDPC code for the multi-source system.

### III. Multi-edge Type LDPC for Multi-source System

#### A. Multi-edge Type LDPC Codes

The principle of multi-edge type LDPC is to introduce more than one edge type to the Tanner graph [4], where the graph ensemble is specified through two polynomials, one associated to variable nodes and the other associated to constraint nodes. The two polynomials are given by

$$v(r, x) = \sum v_b x^b d^b \quad \text{and} \quad \mu(x) = \sum \mu_d x^d, \quad (5)$$

where $d = (d_1, \ldots, d_{n_m})$ is a multi-edge degree and $x = (x_1, \ldots, x_{n_m})$ denotes variables. Similarly, $b = (b_0, \ldots, b_{n_m})$ is a received degree, and $r = (r_0, \cdots, r_{n_m})$ denotes variables corresponding to received distributions. We use $x^d$ to denote $\prod_{i=1}^{n_m} x_i^{d_i}$ and $r^b$ to denote $\prod_{i=0}^{n_m} r_i^{b_i}$. For more details about multi-edge type LDPC codes one can refer to [5].

#### B. Design of Multi-edge Type LDPC

In this subsection, we propose a novel multi-edge type LDPC scheme in the $M - 1 - 1$ system. In the new coding scheme, we do not change the power allocation of each transmitter (i.e. network capacity is not altered), but propose a new coding scheme to approach the network capacity. Fig. 3 compares the proposed scheme with the conventional bilayer LDPC code. In the conventional scheme, the additional $k_{2m}$ check digits at $r_m$ are only produced from $s_m$’s frame and transmitted at $f_m$. However, in the proposed scheme, the extra $k_{2m}$ bits of information are co-produced from all the sources’ frames and randomly distributed in all the frequency bands.
The multi-edge type LDPC code design for an $M - 1 - 1$ system begins with optimizing the lower Tanner graph in Fig. 4 at rate $R^m_1$ and $R^m_2$ for $s_1, \ldots, s_M$, respectively, which follows the conventional methodology of single link LDPC codes. So the lower graph ensemble of the multi-edge type LDPC codes for $s_m$ is represented by

$$v_m(\mathbf{r}, \mathbf{x}) = r_1 \sum_{d_{1,m}=1}^{d_{c,1,m}} v_{[0,1],[d_{1,m}]} x_{d_{1,m}},$$

$$\mu_m(\mathbf{x}) = \sum_{d_{1,m}=1}^{d_{c,1,m}} \mu_{[d_{1,m}]} x_{d_{1,m}},$$

where $[0,1]$ is the vector $\mathbf{b}$, and $[d_{1,m}]$ is the vector $\mathbf{d}$ of $s_m$. For vector $\mathbf{b}$, all variable nodes in the codeword are transmitted through the source-to-relay channel ($b_1 = 1$) at rate $R^m_1$ and there are no punctured variables ($b_0 = 0$) in the codeword. Vector $\mathbf{d}$ contains only one element since there is only one edge type. $d_{1,m}$ is denoted as the degree of variable nodes and check nodes with the maximum value $d_{c,1,m}$ and $d_{c,1,m}$, respectively. The quantity $v_{\mathbf{b},\mathbf{d}}$ is the the number of variable nodes of type $\mathbf{b}$ and $\mathbf{d}$, and $\mu_{\mathbf{d}}$ is the number of check nodes of the type $\mathbf{d}$ in the graph. The code rate of the lower graph ensemble for $s_m$ is

$$R^m_+ = 1 - \sum_{d_{1,m}=1}^{d_{c,1,m}} \mu_{[d_{1,m}]}.$$

So the sum of code rate for the lower graph ensembles for all the sources is

$$R_+ = \sum_{m=1}^{M} R^m_+ = M - \sum_{m=1}^{M} d_{c,1,m} \mu_{[d_{1,m}]}.$$

In the next step, we design the overall graph considering the relay jointly processing all of the sources’ information. Relay $r$ will transmit additional $k_2 = \sum_{m=1}^{M} k_{2,m}$ parity check digits to $d$ within the relay-to-destination channel capacity $\sum_{m=1}^{M} R^m_2$, in which $R^m_2 n$ bits are allocated to $s_m$. So the variable and check nodes’ polynomials for all sources $s_m$ in the overall graph can be written as

$$v_m(\mathbf{r}, \mathbf{x}) = r_1 \sum_{d_{1,m}=1}^{d_{c,1,m}} v_{[0,1],[d_{1,m}]} x_{d_{1,m}},$$

$$\mu_m(\mathbf{x}) = \sum_{d_{1,m}=1}^{d_{c,1,m}} \mu_{[d_{1,m},0,\ldots,0]} x_{d_{1,m}} + \sum_{d_{2,m}=1}^{d_{c,2,m}} \mu_{[0,d_{2,1},\ldots,d_{2,M}]} x_{d_{2,m}}.$$

These relations mean that for the overall graph ensemble, all variable nodes in the codeword are transmitted through the source-to-destination channel ($b_1 = 1$) at rate $R^m_2$, and there are no punctured variables ($b_0 = 0$). Vector $[d_{1,m},d_{2,m}]$ in (5) represents the number of the sockets of the two edge types $\mathcal{E}_{1m}$ and $\mathcal{E}_{2m}$ in a variable node with $d_{1,m}$ and $d_{2,m}$ respectively. Vector $[d_{1,m},0,\ldots,0]$ in (6) represents the number of the sockets of the edge type $\mathcal{E}_{1m}$ in a check node of the lower graph with $d_{1,m}$. Since the check nodes in the lower graph are only connected to $\mathcal{E}_{1m}$, the number of the sockets of other edge types is all zero. Vector $[0,d_{2,1},\ldots,d_{2,M}]$ in the second equation of (9) represents the number of the sockets of the edge types $\mathcal{E}_{21},\ldots,\mathcal{E}_{2M}$ in a check node of the upper graph with $d_{2,1},\ldots,d_{2,M}$. Since the check nodes in the upper graph are not connected to $\mathcal{E}_{1m}$, the number of the sockets of this edge type is zero. Note that for $s_m$ we have

$$R^m_+ = 1 - \sum_{d_{1,m}=1}^{d_{c,1,m}} \mu_{[d_{1,m},0,\ldots,0]} - \sum_{d_{2,m}=1}^{d_{c,2,m}} \mu_{[0,d_{2,1},\ldots,d_{2,M}]}.$$

So the sum of code rate for the overall graph ensemble can be computed as follows.

$$R_+ = M - \sum_{m=1}^{M} \left( \sum_{d_{1,m}=1}^{d_{c,1,m}} \mu_{[d_{1,m},0,\ldots,0]} + R^m_+ \right).$$

We optimize the whole system by regarding all the sources’ frames as a super block, which accesses the relay with code rate $R_+$ for the lower graph, and accesses the destination with code rate $R_-$ for the overall graph. Besides (7), (12) and (13),
there are several constraints that should be satisfied for each $s_m$ as follows:

$$
\psi[0,1],[d_{1,m}]= \sum_{d_{2,m}=0}^{d_{2,m}} \psi[0,1],[d_{1,m},d_{2,m}],
$$

$$
\mu[d_{1,m}] = \mu[d_{1,m},0, \cdots, 0]
$$

$$
\sum_{d_{1,m}=1}^{d_{1,m}} \mu[d_{1,m}]d_{1,m} = \sum_{d_{1,m}=1}^{d_{1,m}} \psi[0,1],[d_{1,m}]d_{1,m}
$$

$$
\sum_{d_{2,m}=0}^{d_{2,m}} \sum_{d_{2,m} \sim d_{2,m}} \mu[d_{2,m}1,d_{2,m}]d_{2,m} = \sum_{d_{1,m}=1}^{d_{2,m}} \sum_{d_{2,m}=0}^{d_{2,m}} \psi[0,1],[d_{1,m},d_{2,m}]d_{2,m}.
$$

To deduce the average extrinsic mutual information of each edge type, we characterize the code ensemble of $s_m$ by the degree distribution $\lambda_{d_{1,m},d_{2,m}}$, for $i = 1, 2$,

$$
\lambda_{d_{1,m},d_{2,m}}^{i,m} = \frac{\psi[0,1],[d_{1,m},d_{2,m}]}{\sum_{d_{1,m}=1}^{d_{1,m}} \sum_{d_{2,m}=0}^{d_{2,m}} \psi[0,1],[d_{1,m},d_{2,m}]d_{1,m}}.
$$

This defines the percentage of $E_{i,m}$ type edges connected to the variable nodes with $d_{1,m}$ edges in $E_{i,m}$, and $d_{2,m}$ edges in $E_{2m}$. We also define another two types of degree distribution. One is $\rho_{d_{1,m}}^{1,m}$, which denotes the percentage of $E_{1m}$ type edges connected to the check nodes in the lower graph with $d_{1,m}$ edges, i.e.,

$$
\rho_{d_{1,m}}^{1,m} = \frac{\mu[d_{1,m}]d_{1,m}}{\sum_{d_{1,m}=1}^{d_{1,m}} \sum_{d_{1,m}=1}^{d_{1,m}} \mu[d_{1,m}]d_{1,m}},
$$

and the other is $\rho_{d_{2,m},d_{2,m}}^{2,m}$, which denotes the percentage of $E_{2m}$ type edges connected to the check nodes with the edge vector $[0, d_{2,1}, \cdots, d_{2,m}]$, i.e.,

$$
\rho_{d_{2,m},d_{2,m}}^{2,m} = \frac{\mu[d_{2,m}1,d_{2,m}]d_{2,m}}{\sum_{d_{2,m}=0}^{d_{2,m}} \sum_{d_{2,m} \sim d_{2,m}} \mu[d_{2,m}1,d_{2,m}]d_{2,m}}.
$$

IV. PERFORMANCE ANALYSIS AND CODE DESIGN

In [2], Density Evolution (DE) is applied to the bilayer LDPC code profile optimization. Due to the fixed degree of the check nodes in both the lower graph and the upper graphs, the complexity of DE is tolerable. However, the fixed degree deteriorates the system performance as mentioned in the introduction. So in the optimization of the multi-edge type LDPC code, we exploit the extrinsic information transfer (EXIT) functions [12, 13, 14] to reduce the code searching complexity. This will likely lead to better code profiles by canceling the constraints on the check nodes degree as in bilayer LDPC code.

We denote the variable nodes set associated with the codeword bits of $s_m$ as $V_m$, and the check nodes set associated with the parity check digits of $s_m$ from the lower graph as $C_{1m}$. The shared check nodes set in the upper graph is denoted as $C_2$. Since $V_m$ is connect to two edge types, i.e., $C_{1m}$ and $C_2$, there are four types of mutual information (MI) defined as follows [14].

$I_{E_{1m}}(1,m)$: The MI between the message sent from $V_m$ to $C_{1m}$ and the associated codeword bit, on each edge in the edge type $E_{1m}$ connecting $V_m$ to $C_{1m}$.

$I_{E_{2m}}(2,m)$: The MI between the message sent from $V_m$ to $C_2$ and the associated codeword bit, on each edge in the edge type $E_{2m}$ connecting $V_m$ to $C_2$.

$I_{E_{1m}}(1,m)$: The MI between the message sent from $C_{1m}$ to $V_m$ and the associated codeword bit, on each edge in the edge type $E_{1m}$ connecting $C_{1m}$ to $V_m$.

$I_{E_{2m}}(2,m)$: The MI between the message sent from $C_2$ to $V_m$ and the associated codeword bit, on each edge in the edge type $E_{2m}$ connecting $C_2$ to $V_m$.

Note that the extrinsic MI on an edge connecting $V_m$ to $C_{1m}$ (or $C_2$), at the output of the variable node, is the a-priori MI for $C_{1m}$ (or $C_2$), i.e., $I_{E_{1m}}(1,m) = I_{Ac}(1,m)$ (or $I_{E_{2m}}(2,m) = I_{Ac}(2,m)$). Similarly, the extrinsic MI on an edge connecting $C_{1m}$ (or $C_2$) to $V_m$, at the output of the check node, is the a-priori MI for $V_m$, i.e., $I_{E_{1m}}(1,m) = I_{Av}(1,m)$ (or $I_{E_{2m}}(2,m) = I_{Av}(2,m)$). Then we have the iterative process as follows.

1. **Variable nodes to check nodes update.** The mean of the extrinsic MI on an edge type $E_{1m}$ connecting $V_m$ to $C_{1m}$, at the output of the variable node in the l-th iteration is

$$
\phi_{V}^{(l)}(1,m) = \sum_{d_{1,m}=1}^{d_{1,m}} \sum_{d_{2,m}=0}^{d_{2,m}} (d_{1,m} - 1) \left[ J^{-1}(I_{Ac}^{(l)}(1,m)) \right]^2 + d_{2,m} \left[ J^{-1}(I_{Ac}^{(l)}(2,m)) \right]^2 + [ J^{-1}(I_{ch}(m)) \right]^2 \lambda_{d_{1,m},d_{2,m}}^{1,m}
$$

(19)

Also, the mean of the extrinsic MI on an edge of $E_{2m}$ connecting $V_m$ to $C_2$, at the output of the variable node in the l-th iteration is

$$
\phi_{V}^{(l)}(2,m) = \sum_{d_{2,m}=1}^{d_{2,m}} \sum_{d_{1,m}=1}^{d_{1,m}} (d_{2,m} - 1) \left[ J^{-1}(I_{Ac}^{(l)}(2,m)) \right]^2 + d_{1,m} \left[ J^{-1}(I_{Av}^{(l)}(1,m)) \right]^2 + [ J^{-1}(I_{ch}(m)) \right]^2 \lambda_{d_{1,m},d_{2,m}}^{2,m}
$$

(20)

We can now get the MI in the l-th iteration as

$$
J_{E_{1m}}^{(l)}(1,m) = J \left( \sqrt{\phi_{V}^{(l)}(1,m)} \right) \text{ and } J_{E_{2m}}^{(l)}(2,m) = J \left( \sqrt{\phi_{V}^{(l)}(2,m)} \right).
$$

2. **Check nodes to variable nodes update.** The update from check node to variable nodes is more complicated. We give the approximation according to [13]. The extrinsic MI on an edge type $E_{1m}$ connecting $C_{1m}$ to $V_m$, at the output of the check node in the l-th iteration is

$$
I_{E_{1m}}^{(l)}(1,m) = 1 - J \left( \sum_{d_{1,m}=1}^{d_{1,m}} (d_{1,m} - 1) \left[ J^{-1}(1 - I_{Ac}^{(l)}(1,m)) \right] \right) \rho_{d_{1,m}}^{1,m}
$$

(21)

The extrinsic MI on an edge type $E_{2m}$ connecting $C_2$ to $V_m$ at the output of the check node in the l-th iteration is

$$
I_{E_{2m}}^{(l)}(2,m) = \sum_{d_{2,m}=0}^{d_{2,m}} \sum_{d_{2,m} \sim d_{2,m}} \mu[d_{2,m}1,d_{2,m}]d_{2,m} = \sum_{d_{1,m}=1}^{d_{2,m}} \sum_{d_{2,m}=0}^{d_{2,m}} \psi[0,1],[d_{1,m},d_{2,m}]d_{2,m}.
$$
more complicated as more than one source participates in the generation of \( C_2 \). We have

\[
I_{E_c}^{(l)}(2, m) = 1 - J \left( \sum_{d_{2,m} = 1}^{d_{2,m}} \sum_{d_{2,m}} \left( d_{2,m} - 1 \right) \left[ J^{-1}(1 - I_{A_c}^{(l)}(2, m)) \right]^2 + \sum_{m' = 1}^{M} d_{2,m} \left[ J^{-1}(1 - I_{A_c}^{(l)}(2, m')) \right]^2 \rho_{0, d_{2,1}, \cdots, d_{2,M}}^{2,m} \right),
\]

(22)

In each iteration process, we make \( I_{A_c}(i, m) = I_{E_c}(i, m) \) and \( I_{A_c}(i, m) = I_{E_c}(i, m) \) for \( i = 1, 2 \). \( I_{ch}(m) \) is determined according to the SNR of the \( m \)-th source-to-destination channel. At the end of the iteration, the MI between the \( V_m \) and the associated codeword is

\[
I(m) = J \left( \sum_{d_{1,m} = 1}^{d_{1,m}} \sum_{d_{2,m}} \left( d_{1,m} - 1 \right) \left[ J^{-1}(1 - I_{A_c}^{(l)}(1, m)) \right]^2 + d_{2,m} \left[ J^{-1}(1 - I_{A_c}^{(l)}(2, m')) \right]^2 \right) \rho_{0, d_{1,1}, \cdots, d_{1,M}}^{2,m} \rho_{0, d_{2,1}, \cdots, d_{2,M}}^{2,m}.
\]

(23)

where

\[
\rho_{\mu, d_{1,1}, \cdots, d_{1,M}}^{2,m} = \frac{v_{\mu}[0,1]\cdot v_{\mu}[1,1]\cdot d_{1,m} + d_{2,m}}{\sum_{d_{1,1}=1}^{d_{1,m}} \sum_{d_{2,1}=0}^{d_{2,m}} \sum_{d_{1,1}=1}^{d_{1,m}} \sum_{d_{2,1}=0}^{d_{2,m}} v_{\mu}[0,1]\cdot v_{\mu}[1,1]\cdot d_{1,1} + d_{2,1} + d_{1,2} + d_{2,2}}.
\]

(24)

According to the design of bilayer LDPC code, we first fix the lower graph codes of all sources, choosing the optimal point-to-point LDPC code to approach capacity \( R^+ \). Then we optimize the overall graph to approach the capacity of the whole system. The code optimization involves finding \( M \) variable node degree distributions, i.e., \( v_{\mu}[0,1]\cdot v_{\mu}[1,1]\cdot d_{1,m} \cdot d_{2,m} \) for \( s_m \) and a check node degree distribution \( \mu[d_{1,1}, \cdots, d_{1,M}] \). The optimization problem is concluded as minimizing the SNR of the whole system, which is a dual problem of maximizing the system threshold. We can therefore write

\[
\text{maximize} \quad \sigma_{sys} = \min \left( \frac{1}{\sigma_1^2 + \cdots + \frac{1}{\sigma_M^2}}, \frac{1}{\sigma_1^2 + \cdots + \frac{1}{\sigma_M^2}} \right),
\]

subject to \( I(m) \rightarrow 1 \), for \( m = 1, \cdots, M \).

V. NUMERICAL RESULTS

We choose the 2-sources case to illustrate the design. The first source has the following capacities: \( R_1^+ = 0.7, R_1^+ = 0.5 \) and \( R_1^+ = 0.2 \). The second source has the following capacities: \( R_2^+ = 0.58, R_2^+ = 0.38 \) and \( R_2^+ = 0.2 \). Applying the proposed method, we get the code profile shown in Table 1. Note that \( \sigma_1^+ \) and \( \sigma_2^+ \) are the thresholds for \( R_1^+ \) and \( R_2^+ \), respectively. Also note that the threshold deduced by EXIT is always larger than that deduced by DE, which has been proved by [16]. So \( \sigma_1^+ \) and \( \sigma_2^+ \) in the Table 1 are larger than the exact values, which are searched by criterion (25).

First of all, we determine the distributions of \( \xi_{E_1} \) and \( \xi_{E_2} \) in the lower graphs for \( s_1 \) and \( s_2 \) to approach the rate \( R^+_1 \) and \( R^+_2 \), respectively. They are designed as the single link LDPC codes and can be directly obtained from [15]. Our main task is to find out the optimal distribution of \( \xi_{E_1} \) and \( \xi_{E_2} \), and the corresponding \( v_{\mu}[0,1]\cdot v_{\mu}[1,1]\cdot d_{1,m} \cdot d_{2,m} \) and \( \mu[d_{1,1}, \cdots, d_{1,M}] \). By adopting the searching criterion of (25) and EXIT curves fitting, we get the three elements with the thresholds \( \sigma_1^+ = 0.970555 \) and \( \sigma_2^+ = 1.189909 \). This is reasonable since \( R_1^+ = 0.58 \), half of which are allocated to \( s_1 \) and the other half are allocated to \( s_2 \). This is reasonable since \( R_1^+ = 0.58 \), half of which are allocated to \( s_1 \) and the other half are allocated to \( s_2 \).

Fig. 5 shows the EXIT charts for the 4 edge types at the system threshold. Fig. 6 and Fig. 7 show the EXIT chart at \( \sigma_1^+ = 0.969555, \sigma_1^+ = 1.19099 \) and \( \sigma_1^+ = 0.971555, \sigma_1^+ = 1.18899, \) respectively. Note that in the three figures, we adopt the same value of the summation of \( \sigma_1^+ \) and \( \sigma_2^+ \). However, the system threshold in Fig. 6 is larger than that of Fig. 5 and the system threshold in Fig. 7 is smaller than that of Fig. 5. So we can see that the EXIT curves in Fig. 6 intersect at a smaller threshold than 1, which means that the iterative decoding at destination will not converge eventually. The EXIT curves in Fig. 7 as well as Fig. 5 intersect at 1, which means the decoding at destination will succeed. Finally, we investigate the separate LDPC code without network coding. In this case, each source is only connected to half of the check nodes at
to the system and execute the EXIT analysis. We conclude in the multi-source scenario. We apply the multi-edge LDPC less extrinsic mutual information from the check nodes at relay s

Fig. 5. EXIT curves for two sources at the system threshold \( \sigma_1^* = 0.970555 \) with rate \( R_1^* = 0.5 \) and \( \sigma_2^* = 1.1899 \) with rate \( R_2^* = 0.38 \).

Fig. 6. EXIT curves with \( \sigma_1^* = 0.969555 \) and \( \sigma_2^* = 1.1909 \).

Fig. 7. EXIT curves with \( \sigma_1^* = 0.971555 \) and \( \sigma_2^* = 1.1889 \).

that each source achieves more extrinsic mutual information due to joint processing at the relay. Therefore, our scheme delivers better performance compared to traditional schemes that do not utilize joint processing at the relay.

VI. CONCLUSION

In this paper, we investigate a network coded LDPC design in the multi-source scenario. We apply the multi-edge LDPC to the system and execute the EXIT analysis. We conclude

REFERENCES

[1] T. J. Richardson, M. A. Shokrollahi, and R. L. Urbanke, “Design of capacity-approaching irregular low-density parity-check codes,” IEEE Trans. Inf. Theory, vol. 47, no. 2, pp. 619-637, Feb. 2001.
[2] P. Razaghi and W. Yu, “Bilayer low-density parity-check codes for decode-and-forward in relay channels,” IEEE Trans. Inf. Theory, vol. 53, no. 10, pp. 3723-3739, Oct. 2007.
[3] M. H. Azimi and J. Yuan, “Design of multi-edge type bilayer-expurgated LDPC codes,” IEEE International Symposium on Information Theory (ISIT), pp. 1988-1992, Jun. 2009.
[4] T. M. Cover and A. A. E. Gamal, “Capacity theorems for the relay channel,” IEEE Trans. Inf. Theory, vol. 25, no. 5, pp. 572-584, Sep. 1979.
[5] T. J. Richardson and R. L. Urbanke, “Multi-edge type LDPC codes,” Available Online: http://lthcwww.epfl.ch/papers/multilinexps.ps.
[6] A. Chakrabarti, A. de Baynast, A. Sabharwal, and B. Aazhang “Low density parity check codes for the relay channel,” IEEE J. of Sel. Areas Commun., vol. 25, pp. 280-291, Feb. 2007.
[7] C. Li, G. Yue, M. A. Kohestapour, X. Wang, and M. Madhianian, “LDPC-coded cooperative relay systems: performance analysis and code design,” IEEE Trans. Commun., vol. 56, no. 3, pp. 485-496, Mar. 2008.
[8] C. Li, G. Yue, M. A. Kohestapour, X. Wang, and M. Madhianian, “LDPC code design for half-duplex cooperative relay,” IEEE Trans. Wireless Commun., vol. 7, no. 11, pp. 4558-4567, Nov. 2008.
[9] R. A. Ahlswede, N. Cai, S. Y. R. Li and R. W. Yeung, “Network information flow,” IEEE Trans. Inf. Theory, vol. 46, no. 6, pp. 1204-1216, Jul. 2000.
[10] X. Bao, J. Li, “Adaptive network coded cooperation (ANCC) for wireless relay networks: matching code-on-graph with network-on-graph,” IEEE Trans. Wireless Commun., vol. 7, no. 2, pp. 574-583, Feb. 2008.
[11] J. Kim, S. Park, J. Kim, Y. Kim, H. Song, “Joint LDPC Codes for Multi-User Relay Channel,” Fourth Workshop on Network Coding, Theory and Applications (NetCod), pp. 1-6, Jan. 2008.
[12] S. Brink, “Convergence behavior of iteratively decoded parallel concatenated codes,” IEEE Trans. Commun., vol. 49, no. 10, pp. 1727-1737, Oct. 2001.
[13] S. Brink, G. Kramer, and A. Ashikhmin, “Design of Low-density parity-check codes for modulation and detection,” IEEE Trans. Commun., vol. 52, no. 4, pp. 670-678, Apr. 2004.
[14] G. Liva and M. Chiani, “Protograph LDPC codes design based on EXIT analysis,” IEEE Global Telecommunications Conference (GLOBECOM), pp. 3250-3254, Nov. 2007.
[15] “A fast and accurate degree distribution optimizer for LDPC code ensembles,” Available Online: http://lthcwww.epfl.ch/research/lpcode/lp.html.
[16] S. R. Kollu and H. Jafarkhai, “On the EXIT chart analysis of low-density parity-check codes,” IEEE Global Telecommunications Conference (GLOBECOM), pp. 1131-1136, Nov. 2005.