International Journal of Innovative Computing, Information and Control
Volume **11**, Number **5**, October **2015**

ICIC International ©2015 ISSN 1349-4198
pp. 1563–1572

A JOINT ITERATIVE DECODE AND EQUALIZATION ALGORITHM FOR LDPC SIGNALS IN DISTRIBUTED MIMO SYSTEMS

Yanyan Huang[1], Yiming Guo[1], Hua Peng[1] and Zheming Lu[2,*]

[1]Information Engineering College
Zhengzhou Information Science and Technology Institute
Zhengzhou 450002, P. R. China
18638575039@163.com

[2]School of Aeronautics and Astronautics
Zhejiang University
No. 38, Zheda Road, Hangzhou 310027, P. R. China
*Corresponding author: zheminglu@zju.edu.cn

Received January 2015; revised May 2015

Abstract. *In cooperative communication systems, frequency synchronization and channel estimation are two important issues required to be addressed in practice. Due to the nature of cooperative communications, multiple frequency offsets may occur and the traditional frequency offset compensations may not apply. To solve this problem, equalization for the time-varying channel has been used in this paper. An efficient joint iterative decode and equalization method was proposed for distributed multiple input multiple output systems, in which different transmitters transmit different low-density parity check code (LDPC) coded signals to the base station. After getting initial estimation values of carrier frequency offsets and channels by the maximum likelihood method, a minimum mean square error equalizer provides the soft symbol information for the LDPC decoder and the decoded information guides the equalizer inversely. By using the iterative process, the demodulation could be realized. Simulation results indicate that the proposed algorithm can be used to obtain a good demodulation performance compared with the existing algorithms.*

Keywords: Distributed multiple input multiple output system, LDPC, Iterative equalization, Decode, Demodulation

1. **Introduction.** If the transmitting antennas are distributed placed in ageographical locations and connected to the same signal processing center by optical fibers or cables, then this kind of multiple input multiple output (MIMO) system can be called distributed MIMO system [1]. The distributed MIMO system can overcome the path loss caused by "near far effect" and "shadow effect" to solve the zero-zone problem in cell communication, and provide a better communication cover rate. Due to the presence of noise and interference channel, the transmission error is unavoidable between transmitted signals and received signals. To reduce the error, channel coding can be carried out at the transmitting end. The low density parity check (LDPC) code is popular in the area of channel coding in recent years [2]. The LDPC [3-5] code is a linear block code with sparse test matrices proposed by Dr. Gallagerin in 1962, which has a better correction performance than the Reed-Solomon (RS) code and Turbo code. Nowadays, many important developments have been achieved in the research on the LDPC theory, such as engineering applications, and implementations of very large scale integration (VLSI) systems [6]. The LDPC code has been applied widely in the areas of optical, satellite, and deep space communication and the fourth generation mobile communication.

Most of the available equalization methods for a distributed MIMO system focus on coded signals. After introducing the linear convolution space time code signals, Wang et al. [7] proposed a minimum mean square error (MMSE) equalizer and a minimum mean square error decision-feedback equalizer (MMSE-DFE) under multiple frequency estimation errors for cooperative relay systems. This method can prevent the inversion of the equalizer matrix per symbol, thereby preventing large dimension matrix operation. In another study [8], an MMSE equalizer based on Kalman filtering was proposed for the same system. However, both abovementioned studies considered the situation that different relays transmit the same signal, whereas different signals are transmitted in real communication. Another study [9] showed the equalizer under multiple frequency offsets, which are induced by multipath. The procedure utilizes the correlation method to estimate multiple frequency offsets and proposes the same equalizer, as mentioned in previous studies [7,8]. Similarly, it is aimed at different path transmission signals of the same. Literature [10] provides the iterative frequency and channel estimation and equalization method based on Turbo code signals. This method also focuses on multipath signals. The frequency and channel estimations are obtained by the pilots and are viewed as the initial values for the MMSE equalizer. After deinterleaving and Turbo decoding, the signals are sent to the parameter estimator and the equalizer, and then the iterative process is operated. In the previous study [11], the authors provide an equivalent band-limited system model and the block MMSE equalizer method by introducing a special 0-1 matrix, and the distributed linear convolution space frequency code signal is proposed. Another study [12] addresses the equalization issue for distributed time-reversal space time block-coded systems when multiple carrier frequency offsets (CFOs) are presented. In that study, a simplified MMSE equalizer is proposed, and this equalizer exploits the nearly banded structure of the channel matrices and utilizes the $\mathrm{LDL}^{\mathrm{H}}$ factorization to reduce the computational complexity.

From the above analysis, the existing studies did not consider the situation that different antennas transmit different coded signals, and most of these studies discussed the space time or linear convolution coded signals. Thus, in our study, we propose a joint iterative decode and equalization method that focuses on the different LDPC-coded signals transmitted by different transmitters in a distributed MIMO system. After obtaining the maximum likelihood (ML) estimation of the frequency offsets and channel coefficients, the MMSE equalizer provides the soft symbol information to the LDPC decoder, and the latter outputs soft information feedback to the equalizer. By using the iterative process, the demodulation of distributed MIMO signals is realized.

The rest of paper is organized as follows. In Section 2, the distributed MIMO signal model is introduced. In Section 3, the joint iterative decode and equalization method is given. Simulation results and performance analysis are shown in Section 4, and conclusions are presented in Section 5.

2. **Distributed MIMO System Model.** In a distributed MIMO system, different users are far apart. Each user has a single antenna, and each antenna sends an independent signal to the receiving base station. Suppose that there are N_t users, one base, and the signal transmitted by each user is $X^{(m)}(n)$, $n = 0, \ldots, N-1$, $m = 1, \ldots, N_t$. The key of the LDPC code is the construction of the low density parity check matrix, which also plays a vital role in the decoding process. According to the structure of different ways, two main kinds of LDPC check matrices are available, namely, random parity check matrix and structured parity check matrix. The former appears more frequently during the early stages of research on the LDPC code. Because of its randomness, the error correction performance is good. However, because the matrix structure of the random parity check

matrix is fixed, simple coding cannot be achieved, and the storage complexity of decoding the parity check matrix is high. Applying the random parity check matrix to an actual system is not easy. The structured parity check matrix is generated through structured algebraic geometry and combination. The coding complexity is small. Thus, this kind of check matrix is easy to achieve. In general, the LDPC encoding method consists of the following steps.

Step 1: The parity check matrix $\boldsymbol{H}$ is generated.

Step 2: The Gaussian elimination method (mod 2 sum between rows) and column transformation are used to transform $\boldsymbol{H}$. The new parity check matrix $\boldsymbol{newH} = [\boldsymbol{I}\ \boldsymbol{P}]$ is generated. The column switching sequence in transformation process is recorded.

Step 3: The new generation matrix $\boldsymbol{newG} = \left[\boldsymbol{P^T}\ \boldsymbol{I}\right]$ is obtained.

Step 4: The formula $\boldsymbol{s^T} = \boldsymbol{x} * \boldsymbol{newG}$ is used to code directly.

Step 5: $\boldsymbol{s^T}$ is transformed in the reverse order of column switching sequence recorded in Step 2. Then, the corresponding codeword $\boldsymbol{s}$ of $\boldsymbol{x}$ is obtained through the LDPC code matrix $\boldsymbol{H}$.

Steps 1 and 2 are implemented by using the method in a previous study [2], and the coded signal $S^{(m)}(n)$, $n = 0, \ldots, N-1$ is obtained by the above procedure. Subsequently, a phase shift keying modulation is performed, and the time domain signal $s^{(m)}(n)$ is as follows:

$$s^{(m)}(n) = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} S^{(m)}(n) e^{j2\pi kn/N}, \quad 0 \leq n \leq N-1 \tag{1}$$

Getting through the effect of flat fading channel, frequency offsets and Gaussian white noise, the received signal at the base station receiver can be written as follows:

$$r(n) = \sum_{m=1}^{N_t} e^{j2\pi f^{(m)} n/N} h^{(m)} s^{(m)}(n) + v(n), \quad n = 0, \ldots, N-1 \tag{2}$$

where $h^{(m)}$ represents the channel response between the m-th user and the base, $f^{(m)}$ is the normalized frequency offset between the m-th user and the base, and $v(n)$ is a complex additive white Gauss noise with zero-mean and variance σ_v^2. Assuming the equalizer length is N_f, the received signal can be rewritten as follows:

$$\tilde{\boldsymbol{r}}_n = \boldsymbol{E}_n \boldsymbol{H} \tilde{\boldsymbol{S}}_n + \tilde{\boldsymbol{v}}_n = \boldsymbol{H}_n \tilde{\boldsymbol{S}}_n + \tilde{\boldsymbol{v}}_n \tag{3}$$

where $\tilde{\boldsymbol{r}}_n = [r(n), \ldots, r(n - N_f + 1)]^T$, $\tilde{\boldsymbol{v}}_n = [v(n), \ldots, v(n - N_f + 1)]^T$ is the noise vector. $\tilde{\boldsymbol{S}}_n = \left[s^{(1)}(n), \ldots, s^{(N_t)}(n), \ldots, s^{(1)}(n - N_f + 1), \ldots, s^{(N_t)}(n - N_f + 1)\right]^T$. If $\boldsymbol{e}_n = \left[e^{j2\pi f^{(1)} n/N}, \ldots, e^{j2\pi f^{(N_t)} n/N}\right]$ and $\tilde{\boldsymbol{h}} = diag\left(\left[h^{(1)}, \ldots, h^{(N_t)}\right]\right)$, then $\boldsymbol{E}_n$ is an $N_f \times N_f N_t$ sized frequency offset matrix as follows:

$$\boldsymbol{E}_n = \begin{bmatrix} \boldsymbol{e}_n & 0_{1\times N_t} & \cdots & 0_{1\times N_t} \\ 0_{1\times N_t} & \boldsymbol{e}_{n-1} & \cdots & 0_{1\times N_t} \\ \vdots & \vdots & \ddots & \vdots \\ 0_{1\times N_t} & \cdots & \cdots & \boldsymbol{e}_{n-N_f+1} \end{bmatrix} \tag{4}$$

where, $\boldsymbol{H}$ is an $N_f N_t \times N_f N_t$ channel matrix as follows

$$\boldsymbol{H} = \begin{bmatrix} \tilde{\boldsymbol{h}} & 0_{N_t\times N_t} & \cdots & 0_{N_t\times N_t} \\ 0_{N_t\times N_t} & \tilde{\boldsymbol{h}} & \cdots & 0_{N_t\times N_t} \\ \vdots & \vdots & \ddots & \vdots \\ 0_{N_t\times N_t} & \cdots & \cdots & \tilde{\boldsymbol{h}} \end{bmatrix} \tag{5}$$

Our purpose is to obtain the uncoded signal $X^{(m)}(n)$, $n = 0, \ldots, N-1$, $m = 1, \ldots, N_t$ according to the received signal $\tilde{\mathbf{r}}_n$ by the proposed algorithm.

3. Joint Iterative LDPC Decoding and Equalization Algorithm. In this section, we give the joint iterative LDPC decoding and equalization algorithm. Our block diagram is shown in Figure 1. The received signal $r(n)$ is transmitted into the equalizer, the frequency offset, and the channel estimator. The output information of the MMSE equalizer is placed into the log-likelihood ratio (LLR) demapping and LDPC decoding block. Then, the LLR mapping feedback is sent to the equalizer. The specific algorithm process is described in detail as follows.

3.1. Initial estimation of the frequency offsets and channels. According to Equation (2), the parameters to be estimated are $f^{(m)}$, $m = 1, \ldots, N_t$ and $h^{(m)}$, $m = 1, \ldots, N_t$. The ML method is utilized, and it has an optimal estimation performance theoretically. According to the previous study [5], the estimator is as follows:

$$\begin{aligned} \hat{f}^{(m)} &= \arg\max_{f} \left| \sum_{n=0}^{N-1} r^*(n) s^{(m)}(n) e^{j2\pi fn} \right|^2 \\ \hat{\boldsymbol{h}} &= \left(\boldsymbol{H}_n^H \boldsymbol{H}_n\right)^{-1} \boldsymbol{H}_n^H \tilde{\mathbf{r}}_n \end{aligned} \tag{6}$$

Here, the pilots and training information are supposed to be known at the receiver. To obtain the initial values of frequency offsets and channels by Equation (6), the MMSE equalizer is applied.

3.2. MMSE equalizer. As we stated in the introduction section, most existing algorithms use MMSE equalization methods, in which the matrix inversion operation should be operated for every symbol interval. Every symbol block can also be taken. Reference [7] pointed out that the complexity of the symbol block equalization method is higher

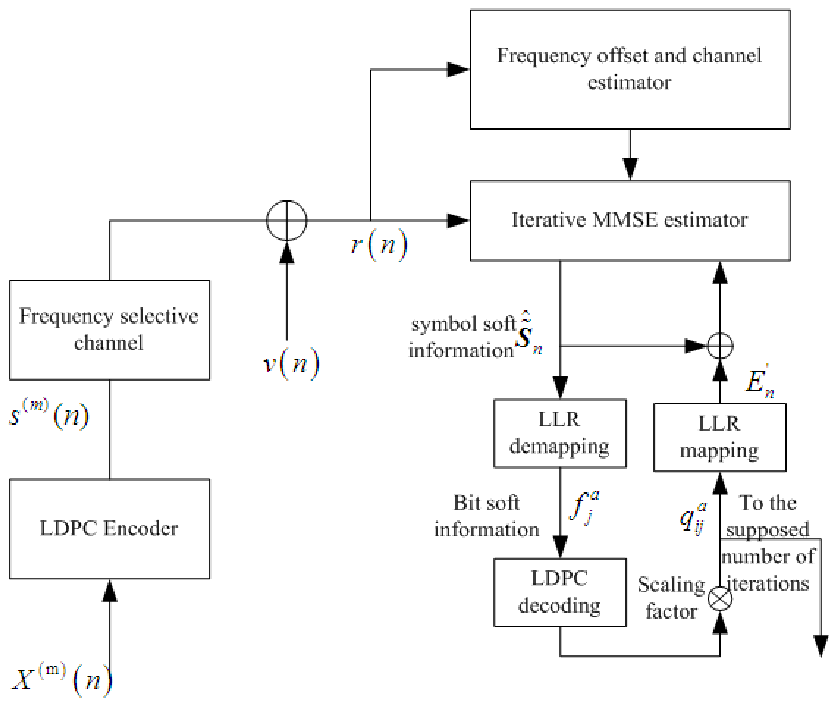


FIGURE 1. The joint iterative LDPC decoding and equalization algorithm

than that of the one operating for every symbol interval, and better performance is not obtained. Thus, the equalization method for every symbol interval is considered in this paper. Thus, the MMSE equalizer can be written as follows:

$$\boldsymbol{f}_n^{MMSE} = \left(\boldsymbol{H}_n\boldsymbol{H}_n^H + c\boldsymbol{I}_{N_f}\right)^{-1}\boldsymbol{H}_n\boldsymbol{i}_D \tag{7}$$

where $c = \sigma_n^2/\sigma_s^2$ is the variance of the symbol sequence, σ_s^2 is the noise variance, and $\mathbf{i}_D$ is an $(N_f + L - 1) \times 1$ vector with 1 at the $(D + 1)$-th element and all 0's elsewhere.

Equation (4) shows that frequency offset matrix is time-varying, which means that matrix inverse should be taken for every symbol interval. Moreover, with increasing signal dimension, the size of the channel matrix dimension becomes increase, and the complexity of matrix inverse becomes higher. Thus, an iterative matrix inverse MMSE equalizer is proposed. This leads to the achievement of the next equalizer coefficient by the previous equalizer coefficient through matrix decomposition. By doing this, the complexity of matrix inverse can be reduced greatly, and a good equalization performance can be obtained.

When $\boldsymbol{R}_n = \boldsymbol{H}_n\boldsymbol{H}_n^H + c\boldsymbol{I}_{N_f}$, at time n and $n+1$, we have:

$$\boldsymbol{R}_n = \begin{bmatrix} \Xi & \boldsymbol{r}_n \\ \boldsymbol{r}_n^H & r_n \end{bmatrix}, \quad \boldsymbol{R}_{n+1} = \begin{bmatrix} r_{n+1} & \boldsymbol{r}_{n+1}^H \\ \boldsymbol{r}_{n+1} & \Xi \end{bmatrix} \tag{8}$$

where Ξ is $(N_f - 1) \times (N_f - 1)$ sub-matrix, $\boldsymbol{r}_n$ and $\boldsymbol{r}_{n+1}$ are $(N_f - 1) \times 1$ vectors, and r_n and r_{n+1} are scalars. Here, $\boldsymbol{R}_n^{-1}$ can be obtained directly by $\boldsymbol{R}_n$ as follows:

$$\boldsymbol{R}_n^{-1} = \begin{bmatrix} \Theta & \boldsymbol{w}_n \\ \boldsymbol{w}_n^H & w_n \end{bmatrix} \tag{9}$$

where $\boldsymbol{\Theta}$ is an $(N_f - 1) \times (N_f - 1)$ sized sub-matrix, $\boldsymbol{w}_n$ is an $(N_f - 1) \times 1$ vector and w_n is a scalar. Thus, we have:

$$\boldsymbol{R}_{n+1}^{-1} = \begin{bmatrix} v_{n+1} + v_{n+1}^2\boldsymbol{r}_{n+1}^H\boldsymbol{\Psi}\boldsymbol{r}_{n+1} & -v_{n+1}\boldsymbol{r}_{n+1}^H\boldsymbol{\Psi} \\ -v_{n+1}\boldsymbol{\Psi}\boldsymbol{r}_{n+1} & \boldsymbol{\Psi} \end{bmatrix} \tag{10}$$

where $v_{n+1} = 1/r_{n+1}$ and

$$\begin{aligned} \boldsymbol{\Psi} &= \Xi^{-1} + \frac{\left(\Xi^{-1}\boldsymbol{r}_{n+1}\right)\left(\Xi^{-1}\boldsymbol{r}_{n+1}\right)^H}{v_{n+1} - \boldsymbol{r}_{n+1}^H\Xi^{-1}\boldsymbol{r}_{n+1}} \\ \Xi^{-1} &= \Theta - \frac{\left(\Theta\boldsymbol{r}_n\right)\left(\Theta\boldsymbol{r}_n\right)^H}{r_n + \boldsymbol{r}_n^H\Theta\boldsymbol{r}_n} \end{aligned} \tag{11}$$

Thus, $\boldsymbol{R}_{n+1}^{-1}$ can be obtained from $\boldsymbol{R}_n^{-1}$. The equalizer coefficient at time $n+1$ can be calculated as follows:

$$\boldsymbol{f}_{n+1}^{MMSE} = \boldsymbol{R}_{n+1}^{-1}\boldsymbol{H}_{n+1}\boldsymbol{i}_D \tag{12}$$

Then, the matrix inverse and the coefficients at all the time can be obtained, and the equalized signal is obtained as $\hat{\tilde{\boldsymbol{S}}}_n = \left(\boldsymbol{f}_n^{MMSE}\right)^H\tilde{\boldsymbol{r}}_n$.

3.3. Iterative equalization and channel parameter estimation.

3.3.1. *LLR demapping.* The soft information is transmitted between the demodulation and decoding as the formation of the bit log-likelihood ratio, namely, the demapping process. MMSE demodulator generates the soft symbol information, which has the following statistical relationship

$$p\left(\hat{s}^{(m)}(n)\left|s^{(m)}(n)\right.\right) = \frac{1}{\left(\sqrt{2\pi\sigma}\right)^2}\exp\left(-\frac{\left\|\hat{s}^{(m)}(n) - s^{(m)}(n)\right\|^2}{2\sigma^2}\right) \tag{13}$$

where $s^{(m)}(n) \in \{s_1, s_2, s_3, s_4\}$ represents the four constellation points. The sending two bits information is $\{00, 10, 01, 11\}$. Taking the first bit $b_{n,1}$ of the sending symbol at time n for example, its LLR is obtained as follows:

$$L\left(b_{n,1}|\hat{s}^{(m)}(n)\right) = \ln \frac{p\left(b_n^1 = 1|\hat{s}^{(m)}(n)\right)}{p\left(b_n^1 = 0|\hat{s}^{(m)}(n)\right)} \tag{14}$$

Combined with the information set of constellation points, we have:

$$L\left(b_{n,1}|\hat{s}^{(m)}(n)\right) = \ln \frac{p\left(s^{(m)}(n) = s_2|\hat{s}^{(m)}(n)\right) + p\left(s^{(m)}(n) = s_4|\hat{s}^{(m)}(n)\right)}{p\left(s^{(m)}(n) = s_1|\hat{s}^{(m)}(n)\right) + p\left(s^{(m)}(n) = s_3|\hat{s}^{(m)}(n)\right)} \tag{15}$$

Utilizing the approximation of $\ln \sum_i \exp(x_i) \approx \max_i(x_i)$ and the priori posteriori probability relationship, Equation (15) can be simplified as follows:

$$L\left(b_{n,1}|\hat{s}^{(m)}(n)\right) \approx \ln \frac{\max\left(\left\|\hat{s}^{(m)}(n) - s_2\right\|^2, \left\|\hat{s}^{(m)}(n) - s_4\right\|^2\right)}{\max\left(\left\|\hat{s}^{(m)}(n) - s_1\right\|^2, \left\|\hat{s}^{(m)}(n) - s_3\right\|^2\right)} \tag{16}$$

From Equation (16), the LLR of the first sending symbol bit at time n can be obtained. Then, the probability of 1 and 0 of this bit after normalizing is as follows:

$$\begin{aligned} p\left(b_{n,1} = 1|\hat{s}^{(m)}(n)\right) &= \frac{1}{1 + L\left(b_{n,1}|\hat{s}^{(m)}(n)\right)^{-1}} \\ p\left(b_{n,1} = 0|\hat{s}^{(m)}(n)\right) &= 1 - p\left(b_{n,1} = 1|\hat{s}^{(m)}(n)\right) \end{aligned} \tag{17}$$

The above formula provides the prior information for the initialization of the decoding process.

3.3.2. *LDPC decoding.* We use the belief propagation (BP) algorithm in LDPC decoding. The BP algorithm is also known as the sum-product algorithm (SPA) [13,14]. "Sum" refers to the check node c_i transfer information r_{ij}^a to adjacent variable nodes z_j ($a = 0, 1$ represents the bit 0 and 1 information respectively), and the summation arithmetic is operated for the information of the other neighboring variable nodes of the check node. "Product" refers to the variable nodes transfer information q_{ij}^a to the adjacent check node, and the production arithmetic is operated for the information of the other neighboring check nodes of the variable node. The exact decoding process is as follows.

Step 1. Initialization.

$f_j^a = p\left(b_{n,i} = a|\hat{s}^{(m)}(n)\right)$, where a, $i \in \{0, 1\}$ is the prior probability initialization value obtained by channel information $\hat{s}^{(m)}(n)$ before updating r_{ij}^a (i.e., the 1 and 0 probability of the check node).

Step 2. Sum.

The updated computing of the information r_{ij}^a transmitted from the check nod c_i to the adjacent variable nodes z_i is as follows:

$$\begin{aligned} r_{ij}^0 &= \frac{1}{2}\left(1 + \prod_{j' \in N(i)/j} \left(1 - 2f_{j'}^1\right)\right) \\ r_{ij}^1 &= \frac{1}{2}\left(1 - \prod_{j' \in N(i)/j} \left(1 - 2f_{j'}^1\right)\right) \end{aligned} \tag{18}$$

where $N(i)/j$ means the remaining variable nods deleting z_j from all variable nods set $N(i)$ adjacent to the check node c_j.

Step 3. Product.

The updated computing of the information q_{ij}^a transmitted from the adjacent variable nodes z_j to the check nod c_j.

$$\begin{aligned} q_{ij}^0 &= k_j f_j^0 \prod_{i' \in M(j)/i} r_{i'j}^0 \\ q_{ij}^1 &= k_j f_j^1 \prod_{i' \in M(j)/i} r_{i'j}^1 \end{aligned} \tag{19}$$

where $M(j)/i$ means the remaining check nods deleting c_j from all check nods set $M(j)$ adjacent to the variable node z_j. k_j is the normalized factor, and $q_{ij}^0 + q_{ij}^1 = 1$. With the first sending bit at time n, we can get the sending bit soft information $q_{n,1}^i$, $i \in \{0, 1\}$ after decoding, and the sending symbol at time n combining the constellation mapping can be written as follows:

$$E_n' = q_{n,1}^0 q_{n,2}^0 s_1 + q_{n,1}^1 q_{n,2}^0 s_2 + q_{n,1}^0 q_{n,2}^1 s_3 + q_{n,1}^1 q_{n,2}^1 s_4 \tag{20}$$

Decoding process produces the soft information to direct the equalization algorithm, the soft information feedback to the equalizer to realize the information transition from decoding to equalizing. In summary, the proposed iterative decoding and equalizing algorithm is given as follows.

Step 1. Taking the frequency offsets and channels estimation as the input of the MMSE equalizer $\hat{f}^{(m)}$, $m = 1, 2, \ldots, N_t$, $\hat{\boldsymbol{h}}$.

Step 2. MMSE equalization. Taking the algorithm introduced in Section 3.2 to equalize $\tilde{r}_n$, thereby getting the symbol soft information $\hat{s}^{(m)}(n)$.

Step 3. Demapping. Demapping the obtained soft information $\hat{s}^{(m)}(n)$ to obtain the bit soft information f_j^a, as the initial prior information of LDPC decoding.

Step 4. LDPC decoding. Decoding the bit soft information in Step 3 to obtain the decoded soft information q_{ij}^a and obtain the symbol soft information E_n'.

Step 5. Iterative decision. If the number of iterations is smaller than the set value, the soft information E_n' is obtained in Steps 4 to 1 to continue the MMSE equalization, where the soft symbol information is updated as follows:

$$\hat{s}^{(m)}(n) = \lambda^d \cdot \left(\boldsymbol{f}_n^{MMSE}\right)^H \tilde{\boldsymbol{r}}_n + \left(1 - \lambda^d\right) \cdot E_n' \tag{21}$$

where d is the number of iteration, and λ is the iterative factor. With increasing number of iterations, the effect of reliable soft decoding information becomes stronger, and the equalization performance improves. If the number of iterations reaches the set value, then go to Step 6.

Step 6. Decoding decision. The each bit likelihood value of the last iteration decoding output soft information q_{ij}^a, let $\lambda_j = \frac{q_{ij}^0}{q_{ij}^1}$ is calculated when making a decision, if $\lambda_j > 1$, $\hat{x}_j = 0$ and if $\lambda_j < 1$, $\hat{x}_j = 1$. When the iterative separation is done, the output is obtained.

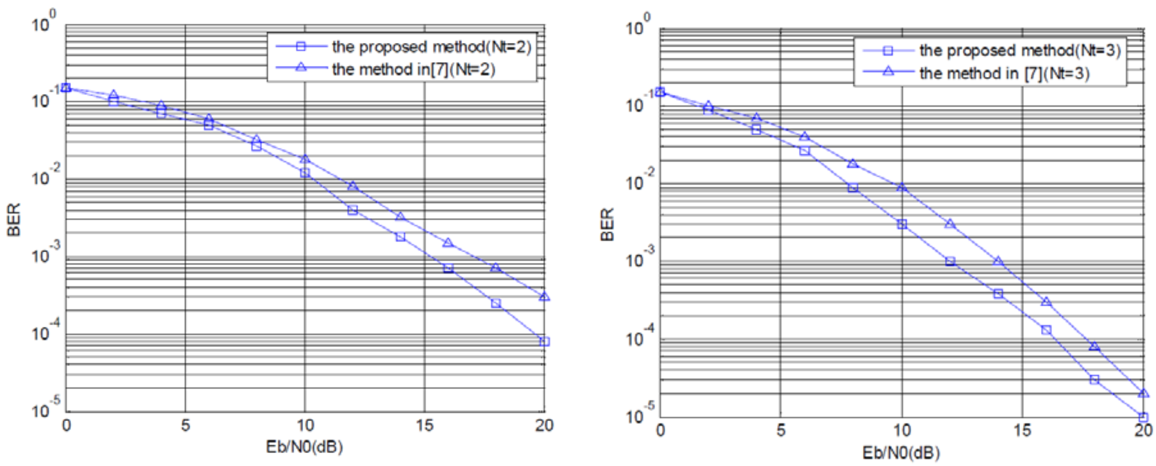
4. **Simulation Results.** In this section, we present some simulation results to illustrate the equalization performance of the proposed iterative decoding and equalization method. The constellation we used is QPSK and the data length is $N = 256$. Assuming that the channel between any user and the base is quasi-static Rayleigh flat fading (channel coefficients are complex Gaussian random variables with zero mean and unit variance), the number of multipath is $L = 9$. The normalized frequency offset value is in the range of $[-0.5, 0.5]$. The range of bit signal-noise-ratio (Eb/N0) is $0 \sim 30$dB, and the Monte Carlo simulation is 1000 times. The simulation uses $(756, 3, 9)$ regular LDPC codes. The code length is 756 bit, for heavy 9, the column weight is 3, the rate is $R = 2/3$, and the parity check matrix H has 252 rows and 756 columns. The bit error rate (BER) vs. Eb/N0

curves are plotted to demonstrate the performance. The lengths of the MMSE equalizer are fixed to 10.

In Test 1, we compare our method with the method in literature [7], where the BER performance of the proposed method versus different users is investigated. The numbers of users are 2 and 3, respectively. All users transmit the same symbols, and CFO is zero. It can be seen from Figure 2 that, the proposed method can achieve a better BER performance than paper [7] in both cases. Because in the previous study [7], the linear convolutive code signal is used, and such signal leads to poorer performance than the LDPC signal in the proposed method. Thus, the BER result is expected. The advantage of LDPC code is also verified. And the BER performance improves when increasing number of users. Due to the fact that the number of equations to be solved increases, the equalization result is improved.

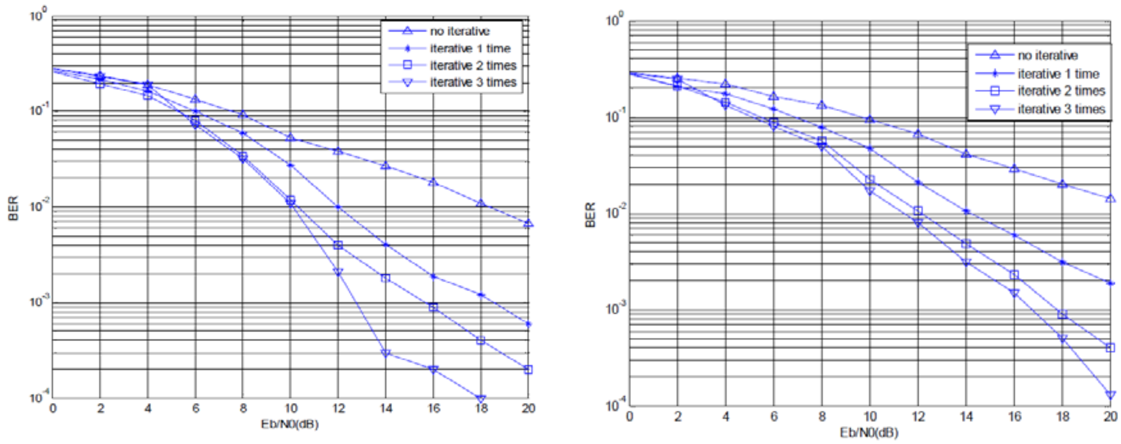
In Test 2, the effect of iteration number on the proposed method versus different users is illustrated. Different users transmit their own independent signal, and CFO is zero. The numbers of users are 2 and 3. The iteration numbers are 0, 1, 2, and 3 respectively. Figure 3 shows that increasing iteration number leads to improvement in BER performance in both cases. When Eb/N0 is lower than 8dB, the improvement of the performance is not so obvious. Because the number of error bit is larger than the LDPC decoding threshold, the decoder is not working. However, at high Eb/N0, the performance improvement is better. When the number of users increases, the unknown parameters to be estimated are increased, and the performance of the proposed method will degrade. When the iterative number is 3, the improvement of the BER performance is small, which means the proposed method can achieve a good performance with few iteration numbers. In the following test, we assume that the iterative number is 3.

In Test 3, we illustrate the equalization performance in Figure 4 when CFOs are presented in the distributed MIMO system. The iteration number is 3. When $Nt = 2$, the normalized CFO is $[0,0][0,0.01][0,0.1][0,0.2]$. When $Nt = 3$, the normalized CFO is $[0,0,0][0,0.01,0.1][0,0.01,0.2][0,0.1,0.2]$. When the normalized CFO increases from 0.01 to 0.1 and 0.2, the BER performance will degrade. This result can be explained by the following: the CFOs will introduce the interference between transmitted signals, and the performance gets lost. Performance with $Nt = 3$ is similar to the performance with



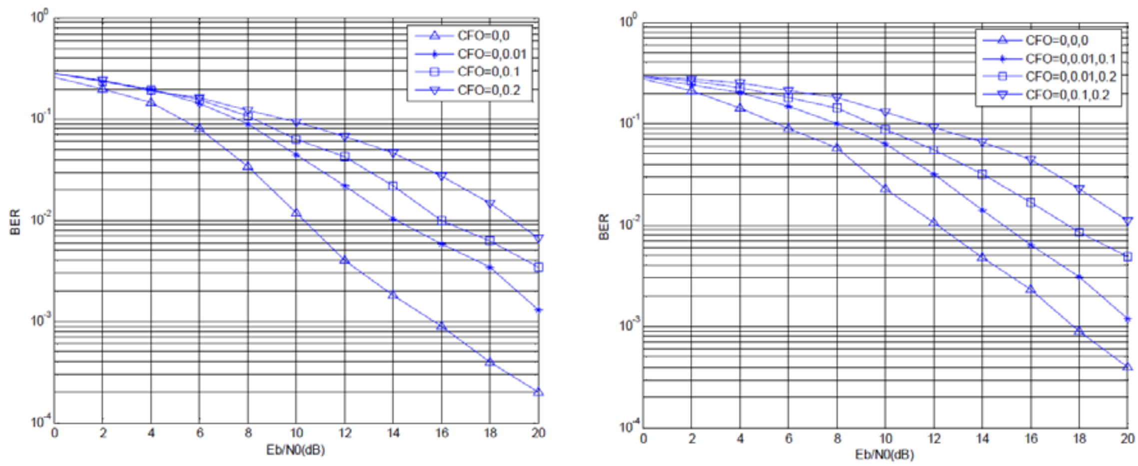
(a) (b)

Figure 2. The BER performance comparison of the proposed method and the method in [7]. (a) two users; (b) three users



(a) (b)

FIGURE 3. The iterative BER performance of the proposed method. (a) two users; (b) three users



(a) (b)

FIGURE 4. The BER performance of the proposed method. (a) two users; (b) three users

$Nt = 2$. The proposed method can lead to good iterative decode and equalization performance under CFOs.

5. **Conclusions.** This paper presents an equalization algorithm for LDPC-coded signals of the distributed MIMO system. The transmitter sending different coding signals is studied. An iterative MMSE algorithm is utilized to avoid the large matrix inverse of every symbol interval. By combining the LDPC decoding, the reliable demodulation can be obtained. Simulation results show that the proposed algorithm can realize the equalization of the distributed MIMO system and has good performance.

REFERENCES

[1] R. Cao and L. Yang, Optimum resource allocation in distributed MIMO systems, *International Journal of Distributed Sensor Networks*, vol.5, no.1, 2009.

[2] G. Zhao, *The Utility Simulation Technology of Spread Spectrum Communication System*, 1st Edition, National Defense Industry Press, Beijing, 2009.

[3] F. Fossorier, Quasi-cyclic low-density parity-check codes from circulant permutation matrices, *IEEE Trans. Information Theory*, vol.50, no.8, pp.1788-1793, 2014.

[4] R. J. Yuan and B. M. Bao, FPGA-based joint design of LDPC encoder and decoder, *Journal of Electronics & Information Technology*, vol.34, no.1, pp.38-44, 2012.

[5] D. J. C. Mackay, Good error correcting cods based on very sparse matrices, *IEEE Trans. Information Theory*, vol.45, no.2, pp.399-431, 1999.

[6] D. J. Costello, J. Hagenauer and H. Imai, Application of error correction codes, *IEEE Trans. Information Theory*, vol.44, no.10, pp.2531-2560, 1998.

[7] H. M. Wang, X. G. Xia and Q. Y. Yin, Computationally efficient equalization for asynchronous cooperative communications with multiple, *IEEE Trans. Wireless Communications*, vol.8, no.2, pp.648-655, 2009.

[8] H. M. Wang, Q. Y. Yin and X. G. Xia, Fast Kalman equalization for time-frequency, *IEEE Trans. Vehicular Technology*, vol.59, no.9, pp.4651-4658, 2010.

[9] S. Ahmed, S. Lambotharan and A. Jakobsson, Parameter estimation and equalization techniques for communication channels with multipath and multiple frequency offsets, *IEEE Trans. Communications*, vol.53, no.2, pp.219-223, 2005.

[10] Q. Yu and S. Lambotharan, Iterative (Turbo) estimation and detection techniques for frequency-selective channels with multiple frequency offsets, *IEEE Signal Processing Letters*, vol.14, no.4, pp.236-239, 2007.

[11] J. Xiao, Y. X. Jiang and X. H. You, A low complexity equalization method for cooperative communication systems based on distributed frequency-domain linear convolutive space-frequency codes, *IEEE Conference on Vehicular Technology*, pp.1-5, 2011.

[12] T. Liu, S. H. Zhu and F. F. Gao, A simplified MMSE equalizer for distributed TR-STBC systems with multiple CFOs, *IEEE Communications Letters*, vol.16, no.8, pp.1300-1303, 2012.

[13] D. L. Zhang, J. Zhang and J. Li, A blind data recovery of PCMA signals based on the turbo iterative processing, *Journal of Wuhan University (Natural Science Edition)*, vol.57, no.5, pp.383-388, 2011.

[14] H. A. Loeliger, An introduction to factors graphs, *IEEE Signal Processing Magazine*, vol.21, no.1, pp.28-41, 2004.