



Belief Propagation-Based Joint Iterative Detection and Decoding Algorithm for Asynchronous IDMA Satellite Systems

Shengfeng Li¹, Senlin Li², Xiang Chen^{2(✉)}, and Ling Wang¹

¹ School of Electronics and Information, Northwestern Polytechnical University, Xi'an 710072, China

sf.li@haige.com, lingwang@nwpu.edu.cn

² School of Electronics and Information Technology, Sun Yat-sen University, Guangzhou 510006, China

{lislin7,chenxiang}@mail.sysu.edu.cn

Abstract. In recent years, a new emerging Interleave-Division Multiple Access (IDMA) developed from the Code-Division Multiple Access (CDMA), with its unique advantages, has been introduced into the satellite communication systems to provide a new multi-user access solution. In IDMA satellite systems, the random distribution of the geographical locations of different user ends (UEs) within the coverage of the satellite ground spot beam will cause serious asynchronous transmission. In this case, the computational complexity of MUD will increase sharply compared to the case of synchronization. Unfortunately, although many simplified algorithms were studied, this problem is still not solved effectively. In this paper, inspired by the successful application of the Belief Propagation (BP)-based joint iterative detection and decoding algorithm in CDMA systems, we adopt this method in asynchronous IDMA satellite systems. Analysis and simulations verify that the BP-based iterative algorithm can effectively reduce the algorithm complexity and has better performance than traditional turbo iterative algorithms in the asynchronous IDMA satellite system.

Keywords: Asynchronous · IDMA · Satellite system · Detection and Decoding · Belief Propagation (BP)

1 Introduction

Interleave-Division Multiple Access (IDMA) is a new emerging multiple access technology based on Code Division Multiple Access (CDMA) proposed by Li

This work was supported in part by the Guangdong Provincial Special Fund For Modern Agriculture Industry Technology Innovation Teams under Grant 2020KJ122, in part by the State's Key Project of Research and Development Plan under Grants 2019YFE0196400, in part by the NSFC under Grant 61771495, and in part by the Guangdong R&D Project in Key Areas under Grant 2019B010158001.

Ping [1]. In the IDMA system, users are distinguished by user-distinct interleavers instead of user-specific spreading sequences. It simplifies the multi-user detection algorithm and improves the system capacity. Considering the advantages of IDMA and the limitations of satellite communications, many studies have shown that IDMA can be used as a new multi-user access technology in satellite communications, and has better performance than CDMA [2,3].

In IDMA system, turbo manner receivers are traditionally used for detection and decoding. These receivers consist of multi-user detection (MUD) modules and decoders with judicious interleaving in between. The complexity of MUD increases in a linear manner with the number of users and iterations. In recent years, researchers have proposed many methods to reduce the complexity of turbo receivers. In the early works of [4], the turbo receivers utilize the Gaussian Chip Detector (GCD) [5] which is also called Elementary Signal Estimation (ESE). In [6], several simplified algorithms based on the ESE are proposed. The Simplified Gaussian Chip Detector (sGCD) with Probabilistic Data Association (PDA) algorithm is introduced to achieve faster convergence of the iterative MUD [7].

However, the above methods are all analyzed in synchronous IDMA systems. Unfortunately, in the actual satellite communication system, the time synchronization of the system is usually based on the Global Positioning System (GPS) or based on the Timing Advance (TA) mechanism [8]. Although this TA mechanism in the random access process achieves the initial uplink synchronization of different users, due to the large transmission distance between the satellite and the ground and the movement of user ends (UEs), the users cannot obtain the TA value fed back by the ground base station in real time. As a result, the system will inevitably have timing deviations. At present, for the geostationary earth orbit (GEO) satellite system, many studies have shown that the timing deviation can be controlled at about 1 ms [8,9]. Therefore, in the IDMA satellite system, due to the problem of timing deviation, the asynchronous delay of the system is usually on the order of several chips. In this case, the complexity of the above turbo iterative detection algorithms increases approximately linearly with the square of the the maximum of delay.

The Belief Propagation (BP)-based algorithm is another classic method to solve the problem of multi-user detection [10]. In asynchronous LDPC-coded CDMA satellite systems, a BP-based joint chip-level iterative detection and decoding algorithm was proposed in [11]. In recent years, LDPC codes have been widely used in satellite mobile communications [12]. LDPC codes can be easily converted into factor graphs and iterative decoding algorithms are usually used. The BP-based joint iterative algorithm constructs a global factor graph with only global iterations to make better use of the iterative structure of LDPC decoders.

In this paper, we apply the BP-based joint iterative detection and decoding algorithm to asynchronous LDPC-coded IDMA satellite systems considering the inheritance of IDMA and CDMA. The aim of this method is to reduce the complexity of the turbo iterative detection algorithms and improve the BER

performance in the asynchronous case. In this article, we have verified the gain of this algorithm through theoretical analyses and simulations under Additive White Gaussian Noise (AWGN) channels.

The rest of this paper is organized as follows. In Sect. 2, the system model for asynchronous IDMA satellite system are described. In Sect. 3, the joint iterative algorithm for IDMA MUD and LDPC decoding is investigated. Complexity analyses of several joint iterative algorithms are given in Sect. 4. In Sect. 5, we evaluate the BER performance of two types of algorithms through simulations. Finally, the conclusion is drawn in Sect. 6.

2 System Description

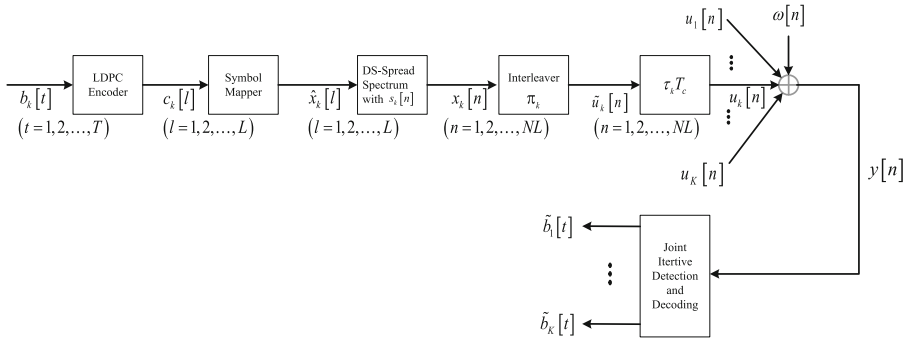


Fig. 1. The asynchronous LDPC-coded IDMA satellite communication system.

In this section, we consider the reverse link of an asynchronous IDMA satellite system, which is an LDPC-coded directsequence binary phase-shift keying (DS-BPSK) IDMA system with K users. In Fig. 1, the k -th user’s information bits are marked as $\{b_k[t]; t = 1, 2, \dots, T\}$ where $k = 1, \dots, K$, N is the length of each information bit block, $n = 0, \dots, N - 1$ and $(\bullet)^T$ denotes transposition. $\{c_k[l]; l = 1, 2, \dots, L\}$ called the coded symbols, where L is the number of coded symbols in a single transmission block of any user after LDPC is encoded. After BPSK modulation, the coded symbols are mapped into modulated symbols with a value space of $\{+1, -1\}$, denoted as $\hat{x}_k[l]$. Assume that $\{s[1], s[2], \dots, s[L_c]; s[m] = \pm 1\}$ is the spreading code of the k -th user; L_c is the code length. After spreading operation, the spreaded symbols which are also called spreaded chips are denoted as $\{x_k[n]; n = 1, 2, \dots, NL\}$. The $\tilde{u}_k[n]$ is the output of the chip-level interleaver π_k . These interleavers π_k are generated independently and randomly.

The signal $u_k[n]$ is delayed by the user specific delay $\tau_k T_c$ where T_c denotes the chip time and τ_k is a nonnegative integer. $D_c = \max_k\{\tau_k + 1\}$ is defined as the maximum total chip delays plus 1 over all users, where $\max_k\{\bullet\}$ says

get the maximum τ_k for all users. Because the uplink channel of satellite mobile communication system is usually described as the AWGN model [13], we consider an asynchronous IDMA system in this case. It is assumed that the signal power of each user is perfectly controlled. The received signal can be described as

$$y[n] = \sum_{k=1}^K h_k[n] \tilde{u}_k[n - d_k] + w[n], \tag{1}$$

where $h_k[n]$ is the effective channel impulse response. $w[n]$ is the additive Gaussian noise with zero mean and the variance σ^2 .

In fact, it is the synchronous IDMA system when the transmission delays $d_k = 0$. Obviously, the existence of asynchronous delay will bring more interference to the system compared to precise synchronization. In this article, we consider the asynchronous model to analyze the robustness of this algorithm.

3 BP-based Joint Iterative Detection and Decoding Algorithm

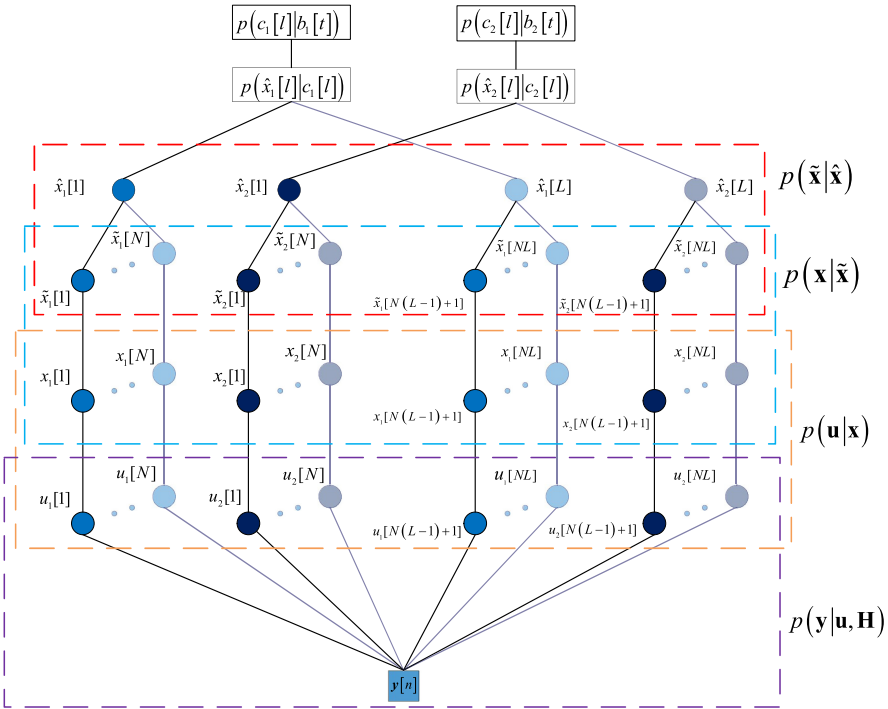


Fig. 2. The factor graph of the IDMA satellite system. For simplification, the number of users $K = 2$.

In this section, we apply a BP-based joint iterative detection and decoding algorithm in asynchronous IDMA systems. We express the joint probability distribution of information bits from all users as a factor graph, and derive the process of information transmission and update in the joint iterative algorithm.

3.1 The Factor Graph Representation of the IDMA System

In the IDMA system, we can decompose the joint probability distribution as

$$\begin{aligned}
 & p(\mathbf{y}, \mathbf{b}, \mathbf{c}, \hat{\mathbf{x}}, \tilde{\mathbf{x}}, \mathbf{x}, \mathbf{u} | \mathbf{H}) \\
 &= p(\mathbf{b})p(\mathbf{y}, \mathbf{c}, \hat{\mathbf{x}}, \tilde{\mathbf{x}}, \mathbf{x}, \mathbf{u} | \mathbf{b}, \mathbf{H}) \\
 &= p(\mathbf{b})p(\mathbf{c} | \mathbf{b})p(\hat{\mathbf{x}} | \mathbf{c})p(\tilde{\mathbf{x}} | \hat{\mathbf{x}})p(\mathbf{x} | \tilde{\mathbf{x}})p(\mathbf{u} | \mathbf{x})p(\mathbf{y} | \mathbf{u}, \mathbf{H}) \\
 &\propto \prod_{1 \leq k \leq K} p(c_k[l] | b_k[l]) \prod_{1 \leq k \leq K} p(\hat{x}_k[l] | c_k[l]) \prod_{1 \leq k \leq K} p(\tilde{x}_k[n] | \hat{x}_k[l]) \\
 &\quad \times \prod_{1 \leq k \leq K} p(x_k[n] | \tilde{x}_k[n]) \prod_{1 \leq k \leq K} p(x_k[n] | \tilde{x}_k[n]) \bar{p}(u_k[n] | x_k[n]) \\
 &\quad \times \prod_{1 \leq n \leq NL} p(y[n] | \mathbf{u}[n], \mathbf{H}[n]),
 \end{aligned} \tag{2}$$

where $\mathbf{H}[n] = [h_1[n], \dots, h_K[n], \dots, h_K[n]]^T$ ($n = 1, \dots, NL$). The vector \mathbf{y} represents the received symbols, \mathbf{b} denotes the vector of the corresponding information bits $\{b_k[t]\}$ ($k = 1, 2, \dots, K$), \mathbf{c} denotes the vector of the corresponding coded symbols $\{c_k[l]\}$ ($k = 1, 2, \dots, K$), $\hat{\mathbf{x}}$ denotes the vector of the modulated symbols $\{\hat{x}_k[l]\}$ ($k = 1, 2, \dots, K$) before spreading, $\tilde{x}_k[n]$ is obtained by only spreading the modulation symbols at the l -th interval $\hat{x}_k[l]$ by the spreading factor N respectively without multiplying spreading sequences, $\tilde{\mathbf{x}}$ is the vector composed of $\tilde{x}_k[n]$, \mathbf{x} denotes the vector of the spread chip $x_k[n]$, \mathbf{u} denotes the vector of the transmitted chip $u_k[n]$.

According to the factorization of the joint probability distribution as (2), we graphically express the received symbols, transmitted chips, spread chips, spreaded modulated symbols, modulated symbols, coded symbols and information bits by different kinds of nodes. The edges denote that these nodes are related. Therefore, we get the factor graph of the IDMA satellite system as shown in Fig. 2. For simplification, the factor graph with $K = 2$ users is given as an example. The nodes in the factor graph are explained as follow.

1. The top nodes $p(c_k[l] | b_k[l])$, $k = 1, \dots, K$, represent the relation between $b_k[l]$ and $c_k[l]$.
2. The nodes on the second layer $p(\hat{x}_k[l] | c_k[l])$, $k = 1, \dots, K$, represent the relation between the $c_k[l]$ and $\hat{x}_k[l]$. We demap the modulated symbols to the coded symbols or reverse by performing the BP algorithm on these nodes.
3. The nodes on the third floor $p(\tilde{x}_k[n] | \hat{x}_k[l])$, represent the relation between the modulated symbols $\hat{x}_k[l]$ and the spreaded modulated symbols $\tilde{x}_k[n]$. By producting every N chips' likelihood ratio on these nodes, we complete the accumulation of the despreading which is multiplied by spreading sequences. N is the spreading factor here. The reason is that the direct spread spectrum operation of modulated symbols in this article includes two processes: 1. Spreading (repeating) the coded symbol sequence according to the spreading factor; 2. Multiplying the spreading sequence. Therefore, in the despreading

process, we will not perform correlation calculations, but will transform it into the following two processes: 1. Multiplying the spreading sequence; 2. Performing accumulation.

4. The nodes in the middle $p(x_k[n]|\tilde{x}_k[n])$, represent the relation between the spread chips $x_k[n]$ and the spreaded modulated symbols $\tilde{x}_k[n]$. We will consider the influence of spreading sequence on the information transmitted by these nodes.
5. The bottom-second nodes $p(u_k[n]|x_k[n])$, represent the relation between the spread chips and the transmitted chips transmitted from each user. We will use interleavers that distinguish users to process the information transmitted on these nodes.
6. Finally, the bottom-most nodes represent $p(y[n]|\mathbf{u}[n], \mathbf{H}[n])$, the relation between the transmitted chips and the received symbols. The observation node $y[n]$ represent the observation function $g_{k,n}(\mathbf{u}[n])$,

$$g_{k,n}(\mathbf{u}[n]) = p(y[n]|u_1[n], \dots, u_K[n]) \propto \exp \left\{ -\frac{\left| y[n] - \sum_{k=1}^K h_k[n]u_k[n] \right|^2}{\sigma^2} \right\}. \tag{3}$$

3.2 Message Passing of Joint Iterative Detection and Decoding Algorithm on the Factor Graph

According to the factor graph, the BP algorithm is adopted to realize the message transfer and update between the variable nodes and the observation nodes. This section will introduce the algorithm process in detail. It is necessary to apply the BP algorithm at the bottom-most nodes to compute $p(y[n]|\mathbf{u}[n], \mathbf{H}[n])$, ($1 \leq n \leq NL$). This process is always called MUD. We use $\mu_{u_k[n] \rightarrow y[n]}^t(u_k[n])$ to indicate that the message is transmitted from the variable node $u_k[n]$ to the observation node $y[n]$ at the t -th iteration. $\mu_{y[n] \rightarrow u_k[n]}^t(u_k[n])$ indicates the message transmission in the opposite direction. Therefore, the message transmission process between $u_k[n]$ and $y[n]$ at the t -th iteration is

$$\mu_{u_k[n] \rightarrow y[n]}^t(u_k[n]) = \mu_{x_k[n] \rightarrow u_k[n]}^{t-1}(u_k[n]) \times \mu_{y[n] \rightarrow u_k[n]}^{t-1}(u_k[n]), \tag{4}$$

$$\begin{aligned} \mu_{y[n] \rightarrow u_k[n]}^t(u_k[n]) &= \sum_{\sim \{u_k[n]\}} g_{k,n}(\mathbf{u}[n]) \prod_{p \in \mathcal{N}(y[n]) \setminus k} \\ &\times \mu_{u_p[n] \rightarrow y[n]}^{t-1}(u_p[n]), \end{aligned} \tag{5}$$

where $\mu_{x_k[n] \rightarrow u_k[n]}^t(u_k[n])$ is obtained by transforming the external probability density function (PDF) feedback from the decoder. Observing the calculation process of Eq. (4), the external information feedback from the decoder $\mu_{\text{dec} \rightarrow \hat{x}_k[l]}^t(\hat{x}_k[l])$ and the information obtained from the previous iteration $\mu_{y[n] \rightarrow u_k[n]}^{t-1}(u_k[n])$ are all included, and the traditional turbo iterative MUD

algorithm only considers the external information feedback from the decoder. Therefore, the BP-based iterative detection algorithm proposed will be more accurate than the turbo iterative detection algorithm, and will have better BER performance.

Observing the formula (5), we find that the iterative update of information using the BP algorithm will lead to very high computational complexity, so it is infeasible to apply it directly to the IDMA system. In order to be able to perform belief-based iteration in the IDMA system more reasonably, we need to simplify formula (5). The appropriate application of the central limit theorem [65] in the BP algorithm enables the above calculation process to be effectively simplified. First, by minimizing the Kullback-Leibler (KL) divergence $KL \left[\mu_{u_k[n] \rightarrow y[n]}^t(u_k[n]) \parallel \hat{\mu}_{u_k[n] \rightarrow y[n]}^t(u_k[n]) \right]$, we can replace the non-Gaussian distribution PDF $\mu_{u_k[n] \rightarrow y[n]}^t$ with the Gaussian distribution PDF $\hat{\mu}_{u_k[n] \rightarrow y[n]}^t$ [14], expressed as:

$$\begin{aligned} \mu_{u_k[n] \rightarrow y[n]}^t(u_k[n]) &\cong \hat{\mu}_{u_k[n] \rightarrow y[n]}^t(u_k[n]) \\ &= \mathcal{N}_c \left(u_k[n], \hat{m}_{u_k[n] \rightarrow y[n]}^t, \hat{v}_{u_k[n] \rightarrow y[n]}^t \right), \end{aligned} \quad (6)$$

where $\hat{m}_{u_k[n]}$ and $\hat{v}_{u_k[n]}$ respectively represent the mean and variance of $\hat{\mu}_{u_k[n] \rightarrow y[n]}^t$, given as follows:

$$\hat{m}_{u_k[n] \rightarrow y[n]}^t = \sum_{\alpha_k \in \{-1, 1\}} \alpha_k \mu_{u_i[n] \rightarrow y[n]}^t(u_k[n] = \alpha_k), \quad (7)$$

$$\hat{v}_{u_k[n] \rightarrow y[n]}^t = \sum_{\alpha_k \in \{-1, 1\}} \left| \alpha_k - \hat{m}_{u_k[n] \rightarrow y[n]}^t \right|^2 \mu_{u_k[n] \rightarrow y[n]}^t(u_k[n] = \alpha_k). \quad (8)$$

Therefore, Eq. (5) can be approximated by some simple linear processes, the process is as follows:

$$\begin{aligned} \mu_{y[n] \rightarrow u_k[n]}^t(u_k[n]) &= \sum_{\sim \{u_k[n]\}} g_{k,n}(\mathbf{u}[n]) \prod_{p \in \mathcal{N}(y[n]) \setminus k} \\ &\quad \times \mu_{u_p[n] \rightarrow y[n]}^{t-1}(u_p[n]) \\ &\cong \int_{\{u_k[n]\}} g_{k,n}(\mathbf{u}[n]) \prod_{p \neq k} \mathcal{N}_c \left(u_p[n], \hat{m}_{u_p[n] \rightarrow y[n]}^t, \hat{v}_{u_p[n] \rightarrow y[n]}^t \right) \\ &\propto \mathcal{N}_c \left(h_k[n] u_k[n], \text{mean}_{y[n] \rightarrow u_k[n]}^t, \text{var}_{y[n] \rightarrow u_k[n]}^t \right), \end{aligned} \quad (9)$$

where $\text{mean}_{y[n] \rightarrow u_k[n]}^t$ and $\text{var}_{y[n] \rightarrow u_k[n]}^t$ respectively represent the mean and variance of $\mu_{y[n] \rightarrow u_k[n]}^t(u_k[n])$, given as follows:

$$\text{mean}_{y[n] \rightarrow u_k[n]}^t = y[n] - \sum_{p \neq k} h_p[n] \hat{m}_{u_p[n] \rightarrow y[n]}^t, \quad (10)$$

$$\text{var}_{y[n] \rightarrow u_k[n]}^t = \sum_{p \neq k} h_p[n] \hat{v}_{u_p[n] \rightarrow y[n]}^t h_p^*[n] + \sigma^2. \quad (11)$$

Traditionally, $\text{mean}_{y[n] \rightarrow u_k[n]}^t$ and $\text{var}_{y[n] \rightarrow u_k[n]}^t$ are computed by summation. Through observation, it is found that a node on the factor graph will receive

different information transmitted from other nodes, and there is a certain correlation between these information. Therefore, we can first calculate the sum of messages including a certain node and all other nodes connected to the node, and then each message can be obtained from the sum by means of a subtraction or an addition corresponding to each node. Therefore, formula (7) and (8) can be simplified as

$$\begin{aligned} \text{mean}_{y[n] \rightarrow u_k[n]}^t &= y[n] - \sum_{p \neq k} h_p[n] \hat{m}_{u_p[n] \rightarrow y[n]}^t \\ &= \text{mean}_{y[n]}^t + h_k[n] \hat{m}_{u_k[n] \rightarrow y[n]}^t, \end{aligned} \tag{12}$$

$$\begin{aligned} \text{var}_{y[n] \rightarrow u_k[n]}^t &= \sum_{p \neq k} h_p[n] \hat{v}_{u_p[n] \rightarrow y[n]}^t h_p^*[n] + \sigma^2 \\ &= \text{var}_{y[n]}^t - |h_k|^2 \hat{v}_{u_k[n] \rightarrow y[n]}^t, \end{aligned} \tag{13}$$

where $\text{mean}_{y[n]}^t = y[n] - \sum_p h_p[n] \hat{m}_{u_p[n] \rightarrow y[n]}^t$, $\text{var}_{y[n]}^t = \sum_p h_p^2[n] \hat{v}_{u_p[n] \rightarrow y[n]}^t + \sigma^2$. Therefore, formula (5) can be approximately expressed as

$$\mu_{y[n] \rightarrow u_k[n]}^t(u_k[n]) \propto \exp\left(-\frac{|h_k u_k[n] - \text{mean}_{y[n] \rightarrow u_k[n]}^t|^2}{\text{var}_{y[n] \rightarrow u_k[n]}^t}\right). \tag{14}$$

The variable node $u_k[n]$ represents the transmission chip after passing through the interleaver of each user and the delay unit, and the node $x_k[n]$ represents the spreaded chip before interleaving. Therefore, the process of message transfer from node $u_k[n]$ to node $x_k[n]$ is expressed as

$$\mu_{u_k[n] \rightarrow x_k[n]}^t(u_k[n]) = \pi_k^{-1}(\mu_{y[n] \rightarrow u_k[n]}^t(u_k[n])), \tag{15}$$

where $\pi_k^{-1}(\cdot)$ represents the de-interleaving operation using the interleaving pattern of the k -th user.

The process of message transfer from $x_k[n]$ to $\tilde{x}_k[n]$ requires the first step of despreading. It is expressed as

$$\mu_{x_k[n] \rightarrow \tilde{x}_k[n]}^t(x_k[n]) = \begin{cases} \mu_{u_k[n] \rightarrow x_k[n]}^t(u_k[n]) & \text{if } s[m] = +1 \\ 1 - \mu_{u_i[n] \rightarrow x_i[n]}^t(u_i[n]) & \text{else.} \end{cases} \tag{16}$$

Further, the process of transferring information from $\tilde{x}_k[n]$ to $\hat{x}_k[l]$ is expressed as

$$\mu_{\tilde{x}_k[n] \rightarrow \hat{x}_k[l]}^t(\tilde{x}_k[n]) = \mu_{x_k[n] \rightarrow \tilde{x}_k[n]}^t(x_k[n]). \tag{17}$$

After the information at the variable node $\hat{x}_k[l]$ is accumulated, the despreading operation is completed and sent to the decoder. The process is as follows:

$$\mu_{\hat{x}_k[l] \rightarrow \text{dec}}^t(\hat{x}_k[l]) = \prod_{n=1+(l-1)N}^{N+(l-1)N} \mu_{\tilde{x}_k[n] \rightarrow \hat{x}_k[l]}^t(\tilde{x}_k[n]). \tag{18}$$

In actual algorithm operation, we usually convert $\mu_{\tilde{x}_k[n] \rightarrow \hat{x}_k[l]}^t(\tilde{x}_k[n])$ into Log-Likelihood Ratio (LLR), so formula (17) can be transformed into the

Algorithm 1. BP-based Joint Iterative Detection and Decoding Algorithm for the IDMA satellite communication system

1: **Initialization**
 2: $t \leftarrow 1$, $\mu_{u_k[n] \rightarrow \mathbf{y}[n]}^t (u_k[n] = 1) \leftarrow 1/2$, $\mu_{u_k[n] \rightarrow \mathbf{y}[n]}^t (u_k[n] = -1) \leftarrow 1/2$,
 $\text{mean}_{\mathbf{y}[n] \rightarrow u_k[n]}^t \leftarrow 0$, $\text{var}_{\mathbf{y}[n] \rightarrow u_k[n]}^t \leftarrow 1$, $\forall k, \forall n$.
 3: **Iterative Update of Messages**
 4: **while** $t \leq \text{MAXITER}$ **do**
 5: **for** $n = 1 \rightarrow NL, k = 1 \rightarrow K$ **do**
 6: **if** $t > 1$ **then**
 7: $\mu_{u_k[n] \rightarrow \mathbf{y}[n]}^t (u_k[n]) \leftarrow \mu_{x_k[n] \rightarrow u_k[n]}^{t-1} (u_k[n]) \times \mu_{\mathbf{y}[n] \rightarrow u_k[n]}^{t-1} (u_k[n])$
 8: **end if**
 9: $\mu_{\mathbf{y}[n] \rightarrow u_k[n]}^t (u_k[n]) \leftarrow \exp \left(- \frac{|h_k u_k[n] - \text{mean}_{\mathbf{y}[n] \rightarrow u_k[n]}^t|^2}{\text{var}_{\mathbf{y}[n] \rightarrow u_k[n]}^t} \right)$
 10: $\mu_{u_k[n] \rightarrow x_k[n]}^t (u_k[n]) \leftarrow \pi_k^{-1} (\mu_{\mathbf{y}[n] \rightarrow u_k[n]}^t (u_k[n]))$
 11: $\mu_{x_k[n] \rightarrow \tilde{x}_k[n]}^t (x_k[n]) \leftarrow \begin{cases} \mu_{u_k[n] \rightarrow x_k[n]}^t (u_k[n]) & \text{if } s[m] = +1 \\ 1 - \mu_{u_k[n] \rightarrow x_k[n]}^t (u_k[n]) & \text{else} \end{cases}$
 12: $\mu_{\tilde{x}_k[n] \rightarrow \hat{x}_k[n]}^t (\tilde{x}_k[n]) \leftarrow \mu_{x_k[n] \rightarrow \tilde{x}_k[n]}^t (x_k[n])$
 13: **end for**
 14: **for** $k = 1 \rightarrow K, l = 1 \rightarrow L$ **do**
 15: $\mu_{\hat{x}_k[l] \rightarrow \text{dec}}^t (\hat{x}_k[l]) \leftarrow \prod_{n=1+(l-1)N}^{N+(l-1)N} \mu_{\tilde{x}_k[n] \rightarrow \hat{x}_k[l]}^t (\tilde{x}_k[n])$
 16: $\mathcal{L}^t (c_k[l]) \leftarrow \log \frac{\sum_{\chi_k^1} \mu_{\hat{x}_k[l] \rightarrow \text{dec}}^t (\hat{x}_k[l])}{\sum_{\chi_k^0} \mu_{\hat{x}_k[l] \rightarrow \text{dec}}^t (\hat{x}_k[l])}$
 17: Decoding the encoded bits of the k -th user, and the decoder outputs the external information $\mu_{\text{dec} \rightarrow \hat{x}_k[l]}^t (\hat{x}_k[l])$
 18: $\mu_{\hat{x}_k[l] \rightarrow \tilde{x}_k[n]}^t (\hat{x}_k[l]) \leftarrow \text{Rep}(\mu_{\text{dec} \rightarrow \hat{x}_k[l]}^t (\hat{x}_k[l]))_N$
 19: **end for**
 20: **for** $n = 1 \rightarrow NL, k = 1 \rightarrow K$ **do**
 21: $\mu_{\tilde{x}_k[n] \rightarrow x_k[n]}^t (\tilde{x}_k[n]) \leftarrow \begin{cases} \mu_{\hat{x}_k[l] \rightarrow \tilde{x}_k[n]}^t (\hat{x}_k[l]) & \text{if } s[m] = +1 \\ 1 - \mu_{\hat{x}_k[l] \rightarrow \tilde{x}_k[n]}^t (\hat{x}_k[l]) & \text{else} \end{cases}$
 22: $\mu_{x_k[n] \rightarrow u_k[n]}^t (x_k[n]) \leftarrow \pi_k (\mu_{\tilde{x}_k[n] \rightarrow x_k[n]}^t (x_k[n]))$
 23: **end for**
 24: $t \leftarrow t + 1$
 25: **end while**

sum operation. In order to facilitate the decoding, we need to convert $\mu_{\hat{x}_k[l] \rightarrow \text{dec}}^t (\hat{x}_k[l])$ into the LLR of the encoding symbol related to the modulation symbol $\hat{x}_k[l]$ as the prior information of the decoder, expressed as

$$\mathcal{L}^t (c_k[l]) = \log \frac{\sum_{\chi_k^1} \mu_{\hat{x}_k[l] \rightarrow \text{dec}}^t (\hat{x}_k[l])}{\sum_{\chi_k^0} \mu_{\hat{x}_k[l] \rightarrow \text{dec}}^t (\hat{x}_k[l])}, \quad (19)$$

where χ_k^1 and χ_k^0 are subsets of modulated symbols. χ_k^1 indicates that the information bit corresponding to each symbol is 1, and χ_k^0 indicates that the information bit corresponding to each symbol is 0.

The external information $\mu_{\text{dec} \rightarrow \hat{x}_k[l]}^t(\hat{x}_k[l])$ output from the decoder is spreaded (repeated) to obtain the information from $\hat{x}_k[l]$ to $\tilde{x}_k[n]$, which is expressed as

$$\mu_{\hat{x}_k[l] \rightarrow \tilde{x}_k[n]}^t(\hat{x}_k[l]) = \text{Rep}(\mu_{\text{dec} \rightarrow \hat{x}_k[l]}^t(\hat{x}_k[l]))_N, \quad (20)$$

where N is the spreading factor, and $\text{Rep}(a)_N$ means that the element “ a ” is repeated N times. Considering the influence of multiplying this information by the spreading sequence, the information from $\tilde{x}_k[n]$ to the variable node $x_k[n]$ is obtained, which is expressed as:

$$\mu_{\tilde{x}_k[n] \rightarrow x_k[n]}^t(\tilde{x}_k[n]) = \begin{cases} \mu_{\hat{x}_k[l] \rightarrow \tilde{x}_k[n]}^t(\hat{x}_k[l]) & \text{if } s[n] = +1 \\ 1 - \mu_{\hat{x}_k[l] \rightarrow \tilde{x}_k[n]}^t(\hat{x}_k[l]) & \text{else.} \end{cases} \quad (21)$$

The information obtained by formula (21) is interleaved to obtain the information transmitted from $x_k[n]$ to $u_k[n]$ as follows:

$$\mu_{x_k[n] \rightarrow u_k[n]}^t(x_k[n]) = \pi_k(\mu_{\tilde{x}_k[n] \rightarrow x_k[n]}^t(x_k[n])), \quad (22)$$

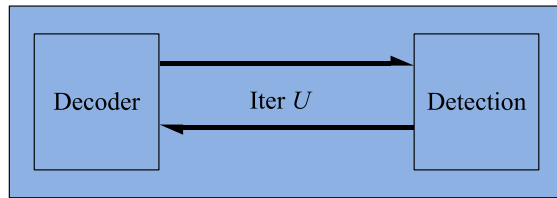
where $\pi_k(\cdot)$ represents the interleaving operation using the interleaving pattern of the k -th user. In summary, the algorithm is given as Algorithm 1.

In the traditional Turbo iterative detection and decoding algorithm, the multi-user detection and decoder form an independent, nested iterative structure. The difference is that in order to make more effective use of the iterative structure of decoding, we unify the iterative process of the decoder and detector based on the BP algorithm to form a joint global iterative structure and optimize the overall iterative structure of the algorithm.

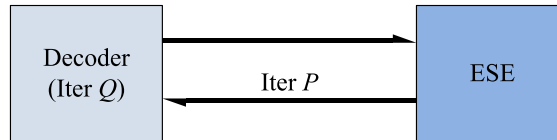
4 Complexity Analysis

Figure 3 shows the different iterative structure diagram of 2 kinds of algorithms. In this paper, we organically unite the iterative process of the decoder and the iterative process of the MUD based on the BP algorithm to form a global iterative structure, and update the soft information of the decoder and MUD at the same time during each global iteration. The schematic diagram of the iterative process is shown in Fig. 3(a). In the turbo detection and decoding algorithm, the iterative process of the decoder is nested in the MUD iterative process independently, that is, the external information required for each iteration of the MUD is fed back by the decoder after Q iterations, its iterative structure is shown in Fig. 3(b). In the IDMA system, the traditional turbo iterative detection and decoding structure consists of an ESE mould and K decoders.

For the fairness of the algorithm complexity comparison, we ensure that the total number of iterations experienced by the LDPC decoder in the two types of algorithms is same, that is, assuming that the global iteration number of the joint iterative algorithm is U , the iteration number of ESE in the turbo iteration structure is P , satisfies $U = P \times Q$. Furthermore, we compare the complexity of the iterative process of the LDPC decoder and the iterative process of the MUD.



(a) Joint Iterative Detection and Decoding Algorithm



(b) Turbo Iterative Detection and Decoding Algorithm

Fig. 3. The iterative structure and complexity comparison.

4.1 Complexity Comparison of LDPC Decoder

In this BP-based joint iterative algorithm, LDPC decoding and MUD form an integral iterative process to run together. However, in the Turbo iterative detection and decoding algorithm, the LDPC decoder operates independently. For the above two types of algorithms, a single user’s single LDPC decoding iteration performs the same operation, that is, obtains the external information of a certain information bit and converts it to the LLR output. For the principle of fairness of comparison, the iteration times of the two types of algorithms are made to satisfy $U = P \times Q$ to ensure that the total iteration times of the LDPC decoder are equal. At this time, the computational complexity of the LDPC decoders in the two types of algorithms are equal.

4.2 Complexity Comparison of Iterative MUD

In asynchronous IDMA satellite systems, we compare the complexity of the MUD in Algorithm 1 with a classical detection algorithm in Section IV of [15] which is equivalent to the sGCD algorithm in Section III.C of [7], and simplified ESE algorithms proposed in [6].

In the iterative process, the BP-based iterative detection algorithm uses the factor graph structure to iteratively update the message for each node. It does not require a large number of matrix operations but is composed of simple linear calculations, so its computational complexity is not subject to the system delay in asynchronous systems. At nodes $y[n]$, for K users, the calculations of (14) per chip per iteration include the computation of $\text{mean}_{y[n] \rightarrow u_k[n]}^t$ and $\text{var}_{y[n] \rightarrow u_k[n]}^t$

expressed by (12) and (13) which needs $O(K^2)$ operations. Other calculation processes in the joint iterative detection algorithm are simple addition or multiplication operations that are not associated with the number of users K . Therefore, the total computational complexity of K users in a single iteration of a single chip is $O(K^2)$.

The MUD of the traditional turbo iterative algorithm requires lots of matrix operations. In asynchronous IDMA systems, the complexity of the algorithm will increase significantly as the system delay increases. The computational complexity of the sGCD algorithm mainly focuses on the calculation of $\sigma_{\zeta_k^{(\ell)}[j]}^2$ in [15]. Since the dimension of matrix \mathbf{H} is $K(2D_c - 1)$, for K users, the calculations of $\sigma_{\zeta_k^{(\ell)}[j]}^2$ per chip per iteration needs $O(K(K(2D_c - 1))^2)$ operations. Therefore, the sGCD's computational complexity of K users in a single iteration of a single chip is $O(K^3 D_c^2)$. As introduced in [6], the two simplified ESE algorithms named ESE.SV1 and the ESE.SV2, which are obtained by simplifying the $\sigma_{\zeta_k^{(\ell)}[j]}^2$ in the sGCD algorithm to a constant. Therefore, the computational complexity of K users in a single iteration of a single chip will be reduced to $O(K^2 D_c^2)$.

Compared with traditional turbo iterative algorithm, the BP-based joint iterative detection and decoding algorithm achieves a significant reduction in computational complexity, and this method avoids complex matrix operations, so its computational complexity will not be affected by the asynchronous delay of the system. It is foreseeable that as the number of users and the asynchronous delay of the system increases, this method will have greater gains in reducing complexity and be more conducive to system implementation.

5 Simulation Results

In our simulations, the number of users $K = 4$, the length of the information bit $T = 1024$ and the spreading factor is $N = 4$. The spreading sequences are derived from [16]. We set the code rate $R_c = 1/4$. BPSK modulation is considered in the our simulation. We assume that user delay can be acquired. We also suppose a perfect power control. The following simulations are achieved in asynchronous IDMA systems.

Without loss of generality, we perform simulations in the GEO satellite system with the TA and without the TA. We set the data transfer rate to 4kbps. Firstly, We accomplished the simulations with the TA in IDMA satellite systems, and the user's chip delays are set as $\tau_k = k - 1$, and Fig. 4 shows the results. Further, we do not perform time synchronization based on the TA mechanism in the system, and the maximum user chip delay of the system is $\tau_k = 168$. The results are shown in the Fig. 5. In the process of comparing algorithm performance, it is usually ensured that the total number of iterations of different algorithms is same to ensure the fairness of the comparison. This means $U = P \times Q$. In the case that the algorithms all meet the convergence, we set $U = 50$, $p = 5$ and the number of iterations of the LDPC decoder $Q = 10$ which is not less than the number of iterations required for LDPC decoding to reach convergence [17].

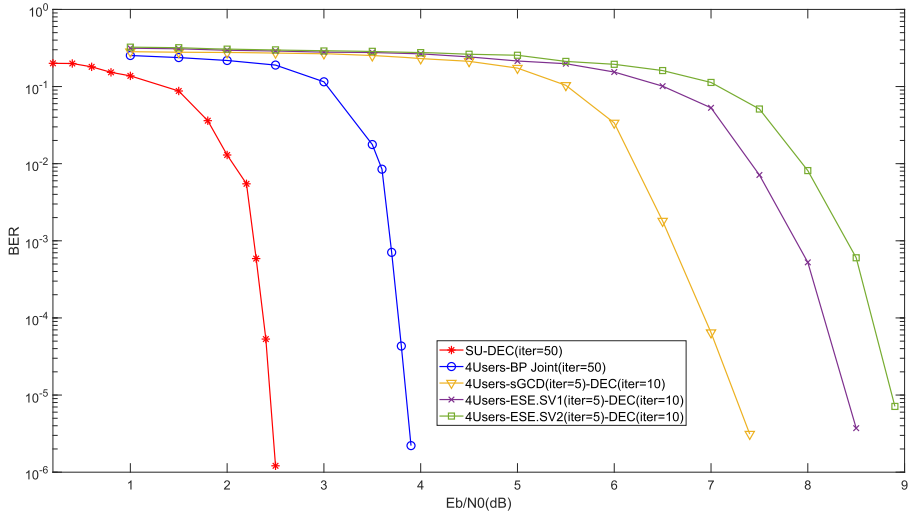


Fig. 4. Performance comparison of the BP-based joint iterative algorithm and turbo iterative detection and decoding algorithms. With TA.

In Fig. 4, the leftmost curve is the performance of the single-user case (SU-DEC(iter = 50)). After 50 iterations, the performance of the BP-based joint iterative algorithm is about 1.4 dB away from that of the SU-DEC. However, at a given BER level of 10^{-6} , the performance of the BP-based joint iterative algorithm outperforms that of the sGCD, ESE.SV1 and ESE.SV2 by 3.5 dB, 4.5 dB and 5 dB respectively. This is due to the MUD of the traditional turbo iterative algorithm only uses the external information fed back by the current iterative decoder to update the LLR, and the external information contained in the received symbol in the previous iteration is lost, resulting in BER performance loss.

Figure 5 shows that the BP-based joint iterative algorithm can save about 4.5 dB, 4.9 dB and 5.4 dB than the sGCD, ESE.SV1 and ESE.SV2 at a given BER level of 10^{-6} with user delay chips get much larger. Comparing the results in Fig. 4, it is obvious that as the asynchronous delay increases, the performance loss of the BP-based joint iterative algorithm is about 0.2 dB, but the performance loss of turbo iterative algorithms exceeds 0.5 dB. The reason is that the information of the previous iteration used by the MUD of the BP-based joint iterative algorithm is the channel information containing the delay information, which reduces the sensitivity of this method to the asynchronous delay.

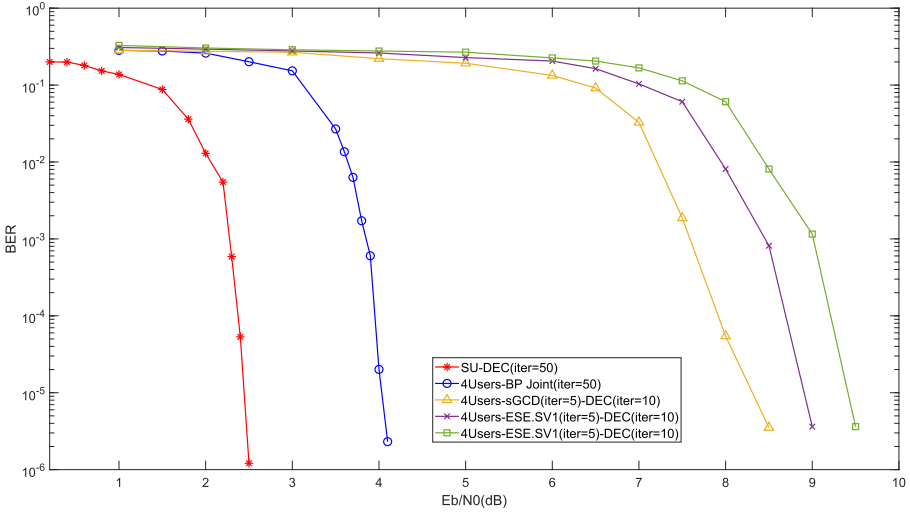


Fig. 5. Performance comparison of the BP-based joint iterative algorithm and turbo iterative detection and decoding algorithms. Without TA.

6 Conclusion

In this paper, we adopt the BP-based joint iterative detection and decoding algorithm in the asynchronous IDMA satellite communication system. This method solves the problem that the computational complexity increases with the asynchronous delay in the traditional turbo method and reduces performance loss. This algorithm breaks the independent iterative structure of the detector and the decoder and forms a global iteration.

On this basis, we established the probability distribution factor diagram of the IDMA satellite communication system, derived the probabilistic message update process using the BP algorithm and simplified it by using the central limit theorem. This algorithm improves the update accuracy of iterative probability information and eliminates the matrix operation in the turbo iterative detection and decoding algorithm, which greatly reduces the computational complexity and makes it immune to system delay. Finally, the simulation results verify that the BP-based joint iterative algorithm has good gains in reducing complexity and improving BER performance in asynchronous IDMA satellite systems.

References

1. Ping, L., Wu, K., Liu, L., Leung, W.: A simple, unified approach to nearly optimal multiuser detection and space-time coding. In: ITW02, Inde (2002)
2. Wang, H., Liu, G.L., Ge, X., Mao, X.P.: Performance evaluation and capacity analysis for IDMA-based satellite communication system. In: 2011 6th International ICST Conference on Communications and Networking in China (CHINACOM), Harbin, pp. 604–608 (2011)

3. Ge, X., Liu, G.L., Zhang, N.: A novel MAC protocol for IDMA-based multi-beam satellite communication systems. *Int. J. Commun. Syst.* **27**(12), 4038–4058 (2014)
4. Ping, L., Liu, L., Leung, W.: A simple approach to near-optimal multiuser detection: interleave-division multiple-access, vol. 1, pp. 391–396 (2003)
5. Liu, L., Ping, L.: Iterative detection of chip interleaved CDMA systems in multipath channels. *Electron. Lett.* **40**(14), 884–886 (2004)
6. Mahafeno, I., Langlais, C., Jego, C.: Cth12-4: reduced complexity iterative multiuser detector for IDMA (interleave-division multiple access) system. In: *Proceedings IEEE GLOBECOM 2006*, pp. 1–5 (2006)
7. Cristea, B., Roviras, D., Escrig, B.: Turbo receivers for interleave-division multiple-access systems. *IEEE Trans. Commun.* **57**(7), 2090–2097 (2009)
8. Kim, H., Kang, K., Ann, D.S.: Uplink timing synchronization for OFDMA based mobile satellite communications, pp. 831–831 (2009)
9. Kim, H., Hong, T.C., Kang, K., Ku, B.J., Kim, S., Yeo, S.: A satellite radio interface for IMT-Advanced system using OFDM. In: *2010 International Conference On Information and Communication Technology Convergence (ICTC)*, pp. 303–308 (2010)
10. Guo, Q., Ping, L.: LMMSE turbo equalization based on factor graphs. *IEEE Sel. Areas Commun.* **26**(2), 311–319 (2008)
11. Gu, N., Wu, S., Kuang, L., Ni, Z., Lu, J.: Belief propagation-based joint iterative algorithm for detection and decoding in asynchronous CDMA satellite systems. *EURASIP J. Wirel. Commun. Netw.* **2013**, 234 (2013)
12. Pei, Y., Liu, C., Feng, H., Shi, Y.: An LDPC-based physical layer solution in satellite interactive system, pp. 425–430 (2011)
13. Weerackody, V., Gonzalez, L.: Performance of satellite communications on the move systems in the presence of antenna pointing errors. In: *IEEE Military Communications Conference, 2006*, Washington, DC, October 2006, pp. 1–7. *IEEE* (2006)
14. Bishop, C.: *Pattern Recognition and Machine Learning*. Springer, New York (2006)
15. Kusume, K., Bauch, G., CDMA and IDMA: iterative multiuser detections for near-far asynchronous communications. In: *IEEE 16th International Symposium on Personal, Indoor and Mobile Radio Communications*, Berlin, vol. 2005, pp. 426–431 (2005)
16. Ping, L., Liu, L., Wu, K., Wk, L.: Interleave-division multiple-access. *IEEE Trans. Wirel. Commun.* **5**, 2869–2873 (2004)
17. Chen, J., Dholakia, A., Eleftheriou, E., Fossorier, M., Hu, X.Y.: Reduced-complexity decoding of LDPC codes. *Commun. IEEE Trans.* **53**(8), 1288–1299 (2005)