1.3.4

The 31st Symposium on Information Theory
and its Applications (SITA2008)
Kinugawa, Tochigi, Japan, Oct. 7–10, 2008

A Note on the Iterative Interference Cancellation and Decoding for Coded CDMA

Shunsuke HORII * Toshiyasu MATSUSHIMA * Shigeichi HIRASAWA *

Abstract— In a code division multiple access wireless communication environment, multiuesr detection (MUD) is an important Interference cancellation technique. Recently, iterative multiuser joint decoding receivers for the system is well-researched. The receivers consist of soft-input soft-output (SISO) multiuser detector and a bank of single-user SISO channel decoders. At each iteration, extrinsic information are computed in detection and decoding stages and are then used as a priori information in the next iteration such as in Turbo decoding. The basic difficulty with the receiver is the complexity of the MUD stages. Most recently, sum-product algorithm based low-complexity SISO multiuser detector have been developed. In this paper, we develop low-complexity iterative multiuser joint decoding receivers based on these sum-product algorithm based multiuser detectors.

Keywords— Coded CDMA, Joint MUD and Decoding, Turbo principle, Sum-product algorithm

1 Introduction

Multiuser detection (MUD) for interference suppression in code-division multiple access (CDMA) communication systems have been studied deeply for a number of years[1]. Prohivitive computational complexity of the optimal multiuser detectors (which is, in general, exponentially proportional to a number of users in the channel[2]) has motivated the study of a number of low-complexity suboptimal multiuser detectors.

Most of the previous work on MUD focused on uncoded CDMA systems. More recently, research has been focused on multiuser detection for coded CDMA systems. One can view the configuration of the system as a serially concatenated code, in which the spectrum spread code is the inner code, and the convolutional code is the outercode. The complexity of optimal detection and decoding is possibly quite high. This complexity can be mitigated by appealing to the turbo principle for decoding concatenated codes noted above. A large number of works have been studied in this direction[3],[4],[5]. The basic building of a turbo multiuser detector are a soft-input soft-output (SISO) multiuser detector and a bank of single-user SISO channel decoders.

As in the case for the uncoded CDMA systems, the basic difficulty with the turbo multiuser detector described above is the prohibitive computational complexity of the MUD stage. One way to mitigate the complexity is to apply suboptimal multiuser detector for uncoded CDMA systems.

* Waseda University Okubo, Shinjuku Tokyo 169-8555 Japan. E-mail: horii@matsu.mgmt.waseda.ac.jp

Recently some low-complexity SISO multiuser detector based on the belief propagation algorithm for the uncoded CDMA system are developed. These detector gives a good performance for some spectrum spread codes. Due to the SISO structure, these detector seems to be a good match with turbo principle. The purpose of this paper is to develop low-complexity iterative multiuser joint decoding receivers based on these sum-product algorithm based multiuser detectors. The rest of the paper is organized as follows. In Section 2, the coded CDMA signal model is presented. In section 3, factor-graph representation for joint decoding and iterative joint decoding schemes are presented. Numerical examples are presented in Section 4.

2 Synchronous CDMA Sytem Model

We consider the CDMA discrete-time synchronous system (see [5] and references therein) described by

$$\boldsymbol{y}_i = \boldsymbol{S}\boldsymbol{x}_i + \boldsymbol{\nu}_i, \qquad i = 1, \ldots, n \tag{1}$$

where $\boldsymbol{y}_i \in \mathcal{R}^N$ is the received signal vector at time i, Corresponding filtered noise vector $\boldsymbol{\nu}_i \sim \mathcal{N}(\boldsymbol{0}, \sigma^2\boldsymbol{I})$ is multivariate Gaussian with $\boldsymbol{0}$ mean and covariance matirx $\sigma^2\boldsymbol{I}$. $\boldsymbol{x}_i = [x_{1,i}\ldots, x_{K,i}]^T \in \{-1,+1\}^K$ is the vector of transmitted user modulation symbols at time i. $S = [\boldsymbol{s}_1, \ldots, \boldsymbol{s}_K]$ is the spreading matrix, where $\boldsymbol{s}_k \in \left\{-1/\sqrt{N}, +1/\sqrt{N}\right\}^N$ denote the normalized unit energy spreading sequence of user k. For the sake of simplicity we assume that the received amplitudes for all user equal 1. It is straight forward to extend the following argument to the case without the amplitude assumption. The integers K,N, and n denote the number of users, the spreading factor (chips per symbol), and the code block length (symbols per block), respectively.

Users send independently encoded information. We assume that the user codewords are aligned in time. Equation defined (1) describes the channel during the transmission of one codeword. We let $\mathcal{C}_k$ denote the code of user k, and we let $\phi_k : F_2^{m_k} \to \{-1,+1\}^n$ denote the encoding function for $\mathcal{C}_k$. The coding rate of user k is $R_k = m_k/n$ bits per symbol. we have

$$\mathcal{C}_k = \{\boldsymbol{x} \in \{-1,+1\}^n : \boldsymbol{x} = \phi_k(\boldsymbol{b}), \quad \forall \boldsymbol{b} \in F_2^{m_k}\} \tag{2}$$

We denote the transmitted codeword of user k by $\boldsymbol{x}^k$, and we let $\boldsymbol{X} \in \{-1,+1\}^{K\times n}$ denote the transmitted code array obtained by stacking by rows the transmitted codewords $\boldsymbol{x}^1, \ldots, \boldsymbol{x}^K$ (the transmitted symbol vector $\boldsymbol{x}_i$ at time i is the ith column of $\boldsymbol{X}$).

3 Graph Representation for Joint Decoding and Iterative Algorithms

In this section factor-graph representation [6] for the joint decoding problem is derived with reference to [5]. By applying the sum-product algorithm to the resulting factor-graph, we obtain a class of iterative decoding algorithms which approximate optimal maximum a posterior marginal (MPM) decoding.

3.1 Factor-Graph Representation

Let $\boldsymbol{b}^k = [b_{k,1}, \ldots, b_{k,m_k}]$ be the vector of information bits of user k, and let $\boldsymbol{Y} = [\boldsymbol{y}_1, \ldots, \boldsymbol{y}_n]$ be the received signal. We assume that the vector channel (1) is memoryless and the user information bits have uniform distribution. Then we can write the posterior distribution of $\{\boldsymbol{b}^1, \ldots, \boldsymbol{b}^K\}$ given $\boldsymbol{Y}$ as

$$P(\boldsymbol{b}^1, \ldots, \boldsymbol{b}^K|\boldsymbol{Y}) \propto \prod_{i=1}^{n} q_i(\boldsymbol{x}_i) \prod_{k=1}^{K} p_k(\boldsymbol{x}^k, \boldsymbol{b}^k) \tag{3}$$

where we define the code constraint functions

$$p_k(\boldsymbol{x}^k, \boldsymbol{b}^k) = 1\left\{\boldsymbol{x}^k = \phi_k(\boldsymbol{b}^k)\right\} \tag{4}$$

and the channel transition functions

$$q_i(\boldsymbol{x}_i) = \exp\left(-\frac{1}{2\sigma^2}\left|\boldsymbol{y}_i - \boldsymbol{S}\boldsymbol{x}_i\right|^2\right). \tag{5}$$

The factor-graph for $P(\boldsymbol{b}^1, \ldots, \boldsymbol{b}^K|\boldsymbol{Y})$ is a graphical representation of the factorization (3) via a bipartite graph. The factor-graph has variable nodes (drawn as a circle) which representing the variables $\{b_{k,j}\}$ and $\{x_{k,i}\}$, factor nodes (drawn as a square) which representing the functions $\{q_k\}$ and $\{p_i\}$, and edge-connectings between variable nodes and factor nodes if and only if the variable is an argument of the function. Fig. 1 shows the factor graph for (3) in the case of $K = 3$, $n = 4$, and $m_1 = m_2 = m_3 = 2$.

Each code constraint function can also be represented as a factor-graph, depending on the code structure. In particular, if code $\mathcal{C}_k$ is a turbo code[7], low-density parity check (LDPC) code[8], etc.,the subgraph formed by the variable nodes $\boldsymbol{b}^k$, $\boldsymbol{x}^k$ and the factor node p_k can be expanded in well-known forms[6].

3.2 The Sum-Product Algorithm

The optimal MPM detection rule minimizing the average BER for each user code bit is given by

$$\hat{x}_{k,i} = \begin{cases} 1 & \text{if } LLR[x_{k,i}] > 0 \\ -1 & \text{if } LLR[b_{k,k}] < 0 \end{cases} \tag{6}$$

where $LLR[x_{k,i}]$ is the posterior Log Likelihood Ratio (LLR) of bit $x_{k,i}$, given by

$$LLR[x_{k,i}] = \log\frac{P(x_{k,i} = 1|\boldsymbol{Y})}{P(x_{k,i} = -1|\boldsymbol{Y})} \tag{7}$$

Brute force computation of the $LLR[x_{k,i}]$ requires complexity of the order of $\prod_{k=1}^{K} |\mathcal{C}_k|$.

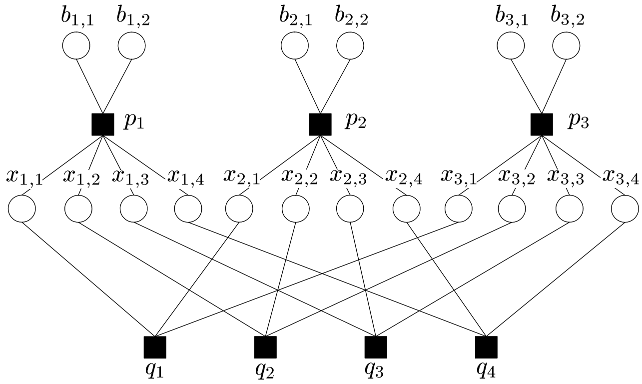

Figure 1: An example of the factor-graph for a multiuser coded CDMA system with $K = 3$ users and code block length $n = 4$.

One way to approximate (7) is applying the sum-product algorithm [6] to the factor graph. Sumproduct algorithm consists of messages which is exchanged by adjacent nodes in the factor-graph.

Let $\Lambda_1[x_{k,i}]$ and $\Lambda_2[x_{k,i}]$ denote the Log Likelihood Ratio (LLR) of $x_{k,i}$ computed in q_k and p_k, respectively. By applying the sum-product computation rules, we obtain follows.

Computation at the channel transition function nodes:

$$\Lambda_1^{(t)}[x_{k,i}] = \log\frac{\sum_{\boldsymbol{x}_i : x_{k,i}=1} q_i(\boldsymbol{x}_i)\prod_{l=1}^{K} P_{l,i}^{(t-1)}(x_{l,i})}{\sum_{\boldsymbol{x}_i : x_{k,i}=-1} q_i(\boldsymbol{x}_i)\prod_{l=1}^{K} P_{l,i}^{(t-1)}(x_{l,i})}, \tag{8}$$

Computation at the code constraint function nodes:

$$\Lambda_2^{(t)}[x_{k,i}] = \log\frac{\sum_{\boldsymbol{x}\in\mathcal{C}_k, x_{k,i}=1}\prod_{j=1}^{n} Q_{k,j}^{(t-1)}(x_{k,j})}{\sum_{\boldsymbol{x}\in\mathcal{C}_k, x_{k,i}=-1}\prod_{j=1}^{n} Q_{k,j}^{(t-1)}(x_{k,j})} \tag{9}$$

where $P_{l,i}(x_{l,i})$ and $Q_{k,j}(x_{k,j})$ are defined as

$$P_{l,i}^{(t-1)}(x_{l,i}) = \frac{1}{2}\left[1 + x_{l,i}\tanh\left(\frac{1}{2}\Lambda_2^{(t-1)}[x_{l,i}]\right)\right] \tag{10}$$

$$Q_{k,j}^{(t-1)}(x_{k,j}) = \frac{1}{2}\left[1 + x_{k,j}\tanh\left(\frac{1}{2}\Lambda_1^{(t-1)}[x_{k,j}]\right)\right] \tag{11}$$

After some iterations $\Lambda_1^{(t)}[x_{k,i}]$ or $\Lambda_1^{(t)}[x_{k,i}]$ are used as approximation to the $LLR[x_{k,i}]$ The information bit detection is obtained by using the above approximated APP.

3.3 Computation of Extrinsic Information

The quantity defined in (9) is the "extrinsic information" (EXT) of the error correcting decoder [7]. It is well known that if the user codes is a turbo code, LDPC code, etc., the approximate computation of (9) carried out efficiently by the sum-product algorithm.[6]

The quantity defined in (8) is the EXT of the optimal multiuser detector [2] with a prior distribution $\{P_{k,i}(x) : k = 1, \cdots, K\}$. In general, the CDMA vector channel (1) has no particular structure, the computational complexity of (8) is $O(2^K)$. Several low-complexity algorithms are derived in [4], [5].

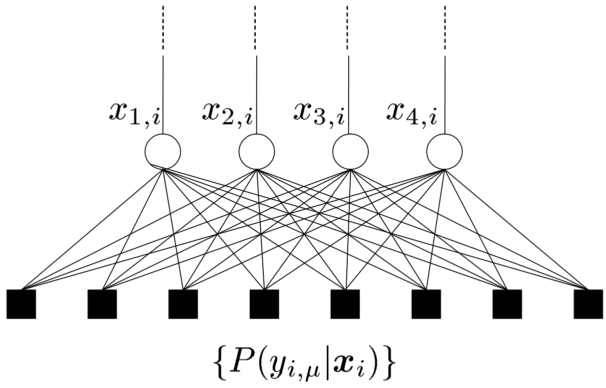

Figure 2: Example of the factor graph for the factorization (13)

Recently, several types of the sum-product algorithm based MUD algorithms have been developed [11], [12], [13]. Since these algotihm are based on the sum-product algorithm, they are the SISO multiuser detector and are good match with turbo principle. In this paper, we deal with these algorithms. Provided the prior LLRs which are computed by channel decoder, several SISO multiuser detector outputs posterior LLRs and thed fed back to the channel decoder.

Sum-product with Hard bit:

In [11], [12], (5) is factorized as follows:

$$q_i(\boldsymbol{x}_i) = \prod_{\mu=1}^{N} P(y_{i,\mu}|\boldsymbol{x}_i) \tag{12}$$

$$P(y_{i,\mu}|\boldsymbol{x}_i) = \exp\left\{-\frac{1}{2\sigma^2}\left(y_{i,\mu} - \sum_{k=1}^{K} s_{k,\mu}x_{k,i}\right)^2\right\} \tag{13}$$

where, $y_{i,\mu}$ and $s_{k,\mu}$ are the μ th componet of the vector $\boldsymbol{y}_i$ and $\boldsymbol{s}_k$, respectively. Based on the above factorization, we can expand the factor node q_i. Example of the expanded factor graph is depicted in Fig. 2.

Applying the sum-product algorithm for the above factor graph, we can obtain following algorithm for the detection for $\boldsymbol{x}_i$,
$\tau = 1, 2, \cdots, T_2$

$$n_{\mu,k}^{(\tau)} = \tanh\left(\sum_{\nu\neq\mu} \tanh^{-1} m_{\nu,k}^{(\tau-1)} + \frac{1}{2}\Lambda_2^{(t)}[x_{k,i}]\right) \tag{14}$$

$$m_{\mu,k}^{(\tau)} = \frac{\sum_{\boldsymbol{x}} x_{k,i} P(y_{i,\mu}|\boldsymbol{x}_i) \prod_{l\neq k}\left(\frac{1+x_{l,i}n_{\mu,k}^{(\tau)}}{2}\right)}{\sum_{\boldsymbol{x}} P(y_{i,\mu}|\boldsymbol{x}_i) \prod_{l\neq k}\left(\frac{1+x_{l,i}n_{\mu,k}^{(\tau)}}{2}\right)} \tag{15}$$

and approximate computation of (8) reduces

$$\tilde{\Lambda}_1^{(t)}[x_{k,i}] = 2\left(\sum_{\mu=1}^{N} \tanh^{-1} m_{\mu,k}^{(T_1)}\right) + \Lambda_2^{(t)}[x_{k,i}] \tag{16}$$

(We use τ for time index of the multiuser detection algorithm to distinguish that of the joint decoding. T_1 denotes the total number of iteration for multiuser detection.) We can discribe the algorithm above, however, it is not practical since the algorithm still requires exponential complexity order in number of users. The basic idea proposed in [11] is to use hard bits instead of using soft bits. Resulting algorithm is described as follows: (See [11] for detail.)
$\tau = 1, 2, \cdots, T_2$

$$\tilde{x}_{k,i}^{(\tau)} = \mathrm{sign}\left[\frac{2}{\sigma^2}\left(h_{k,i} - \sum_{l\neq k} W_{kl}\tilde{x}_{l,i}^{(\tau-1)}\right) + \Lambda_2^{(t)}[x_{k,i}]\right] \tag{17}$$

where

$$h_{k,i} = \boldsymbol{s}_k^T \boldsymbol{y}_i, \qquad W_{kl} = \boldsymbol{s}_k^T \boldsymbol{s}_l. \tag{18}$$

and (16) reduces to

$$\Lambda_1^{(t)}[x_{k,i}] = \frac{2}{\sigma^2}\left(h_{k,i} - \sum_{l\neq k} W_{kl}\tilde{x}_{l,i}^{(T_1)}\right) + \Lambda_2^{(t)}[x_{k,i}] \tag{19}$$

Sum-product with Gaussian approximation:

Kabashima proposed that to approximate the sum-product algorithm based on the gaussian approximation [12]. Kabashima assumed that $\sum_{l\neq k} s_{l,\mu}x_{l,i}$ behave as gaussian variable. Resulting update rule is described as (see [12] for detail)

$$m_{\mu,k}^{(\tau)} = \tanh\left[A^{(\tau)}\left(y_{i,\mu}s_{k,\mu} - \sum_{l\neq k} s_{k,\mu}s_{l,\mu}n_{\mu,l}^{(\tau)}\right)\right] \tag{20}$$

where

$$A^{(\tau)} = \frac{1}{\sigma^2 + \frac{K}{N}(1 - Q^{(\tau)})} \tag{21}$$

$$Q^{(\tau)} = \frac{1}{K}\sum_{k=1}^{K}\left(\tanh\sum_{\mu=1}^{N}\tanh^{-1} m_{\mu,k}^{(\tau)}\right)^2 \tag{22}$$

Sum-product IC[13]:

Recently we developed another type of sum-product algorithm based multiuser detection algorithm in [13]. The RHS of (5) can be factorized as follows

$$q_i(\boldsymbol{x}) \propto \prod_{k=1}^{K} \exp\left(\frac{x_{k,i}h_{k,i}}{\sigma^2}\right) \prod_{k<l} \exp\left(-\frac{x_{k,i}x_{l,i}W_{kl}}{\sigma^2}\right) \tag{23}$$

Based on the above factorization, we can expand the factor node q_i and we can obtain the approximate value of (8) by applying the sum-product algorithm to the graph. Expantion for the previous example is expressed in Fig. 3 The computational complexity of the algorithm is $O(K^3)$ and therefore it is practical. Algorithm detail is omitted here and see [13] for detail. The expanded factor graph seemingly has many cycles, however, the number of cycles will descend for some cases.

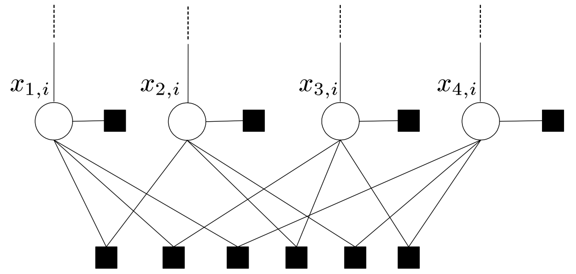

Figure 3: Example of Factor graph expantion of the CDMA vector channel.

If the cross correlation value $\boldsymbol{s}_k^T \boldsymbol{s}_l$ equals to 0, we can eliminate corresponding factor node. Even if the cross correlation value not equal 0, we have confirmed that the sum-product IC work well if the absolute value is small.

3.4 Scheduling

In general, if the factor-graph has any cycles, the sum-product algorithm performance is sensitive to the order in which computation is carried out through the nodes (scheduling). In this paper, we consider the following type of scheduling:

(1) Applying the sum-product algorithm for the factor-graph of $p_k, k = 1, \ldots, K$. We denote the total number of iteration in this phase as T_1

(2) Applying the sum-product algorithm for the factor-graph of $q_i, i = 1, \ldots, n$. We denote the total number of iteration in this phase as T_2

(3) Repeat (1) and (2) for some numbers. We denote the number of sets as T_3

4 Simulations

In this section, we present some simulation examples. Simulation condition is described as follows:

- $K = 7$, $N = 7$, $n = 114$
- All user employ the same rate 1/2 (3,6)-regular LDPC codes
- Shifted version of M-sequence [10] are used as Spreading sequences.

Fig.4 shows the bit error rate curves of the detectors. Number of iteration are set as $T_1 = 10$, $T_2 = 5$, $T_3 = 10$. We can see that sum-produt IC based joint decoding reciever outperform other recievers. This is similar result for uncoded system.

Fig.5 shows the bit error rate curbes of the sum-product IC based joint decoding reciever with different number of iterations. Total number of iteration, i.e., $(T_1+T_2)\times T_3$ is fixed to 20 and we cange the proportion of these value. Result shows that if the total number of iteration is limited, we can get the good detector by increasing the number of iterations at the decoding and detection stage.

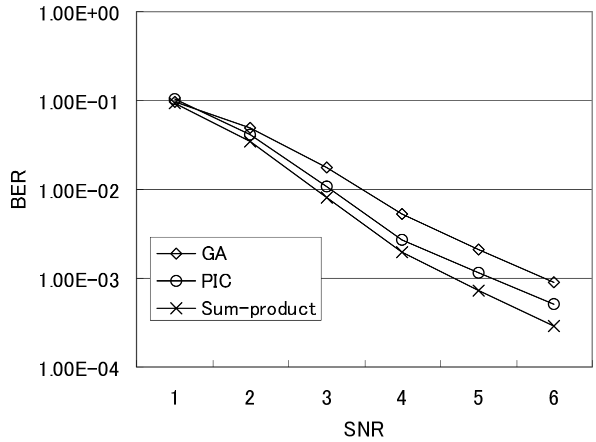

Figure 4: Bit error rate curves of the detectors

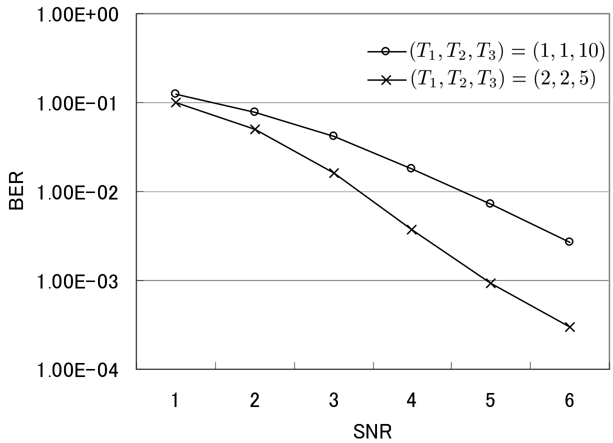

Figure 5: Bit error rate curves of the sum-product IC detectors with different number of iterations

5 Conclusion

In this paper, we have developed a low-complexity iterative receiver for decoding multiuser information data in a coded DS-CDMA system. Simulation results demonstrate that sum-product based multiuser detector shows very good performance. We also consider the scheduling of the algorithm. Accoding to the numerical example, we should update the message serially, rather parallely.

Acknowledgment

One of the authors, Shunsuke Horii, would like to acknowledge all members of Matsushima Lab. and Hirasawa Lab. in Waseda Univ. for their helpful suggestions to this work. This research is partially supported by Category (C) No.15560338 of Grant-in-Aid for Scientific Research, Japan Society for the Promotion of Science.

References

[1] S. Verdú, *Multiuser Detection.* Cambridge, U.K.: Cambridge Univ. Press, 1998

[2] S. Verdú, "Minimum Probability of error for asynchronous Gaussian multiple-access channels," IEEE Trans. Inf. Theory, vol.32, No.1, pp.85-96, January, 1986.

[3] P. Alexander, A. Grant, and M. Reed, "Iterative detection in code-division multiple access with error control coding," European Trans. Telecommun., vol. 9, no. 5, pp. 419-425, Sept. 1999.

[4] X. Wang and V. Poor, "Iterative (turbo) soft interference cancellation and decoding for coded CDMA," IEEE Trans. Commun., vol. 47, pp. 498-519, Feb. 2001.

[5] J. Boutros, G. Caire, "Iterative Multiuser Joint Decoding: Unified Framework and Asymptotic Analysis," IEEE Trans. Inf. Theory, vol. 48, No. 7, July, 2002.

[6] F. R. Kschischang, B. J. Frey and H. A. Loeliger, "Factor Graphs and the Sum-Product Algorithm," IEEE Trans. Inf. Theory, Vol.47, No.2, pp.498–519, February, 2001.

[7] C. Berrou and A. Glavieux, "Near optimum error-correcting coding and decoding: Turbo codes," IEEE Trans. Commun., vol. 44, Oct. 1996.

[8] D. J. C. MacKay, "Good error-correcting codes based on very sparse matrices," IEEE Trans. Inf. Theory, Vol.45, No.2, pp.399–431, March, 1999.

[9] M. K. Varanasi and B. Aazhang, "Multistage detection in asynchronous code-division multiple-access communications," IEEE Trans. Commun. Vol.38 No.4, pp.509–519, April, 1990.

[10] K. W. Yip and T. S. Ng, "Code phase assignment - A technique for high capacity indoor mobile DS-CDMA communications," in IEEE *Vehicular Technology Conf. VTC'94,* pp.1586–1590, June, 1994.

[11] T. Tanaka and M. Okada, "Approximate Belief Propagation, Density Evolution, and Statistical Neurodynamics for CDMA Multiuser Detection," IEEE Trans. Inf. Theory, Vol.51, No.2, pp.700–706, February, 2005.

[12] Y. Kabashima, "A CDMA multiuser detection algorithm on the basis of belief propagation," J. Phys. A: Math. Gen., vol. 36, no. 43, pp.11111-11121, 2003.

[13] S. Horii, T. Suko, T. Matsushima, S. Hirasawa, "Multiuser Detection Algorithm for CDMA based on the Belief Propagation Algorithm," 10th International Symposium on Spread Spectrum Techniques and Applications (ISSSTA2008), Bologna, Italy, August 2008.