# Two-User Turbo-Hadamard Code for Synchronized Gaussian Multiple-Access Channel

Li Xiong * Jun Cheng * Yoichiro Watanabe *

*Abstract*—**Two-user turbo-Hadamard code and its decoding procedure are proposed for synchronized Gaussian multiple-access channel. The encoder of each user consists of 2 component encoders concatenated parallel, and the component vector in the output of one user's encoder is orthogonal to that of other user's encoder. The orthogonality is used to separate data of different users. In iterative decoding, the posterior probability of each user is calculated separately according to its trellis stage thanks to the orthogonality. The total posterior probability is combined by the two posterior probability of users. Then the total extrinsic probability is obtained from the total posterior probability and is passed to other decoder as prior probability.**

***Keywords:* *Turbo-Hadamard code, multi-user code, Gaussian multiple-access channel, decoding.***

## 1 Introduction

We propose a two-user turbo-Hadamard coding for a synchronized Gaussian multiple-access channel (GMAC), where the error correction and users' data separation can be dealt with jointly.

A two-user multiple-access system over GMAC is shown in Fig. 1. In the system, two encoders of users are connected to GMAC. Symbol synchronization is assumed in the channel inputs. The output of the two-user GMAC channel is

$$Y = \sum_{i=1}^{2} X_i + z, \qquad X_i \in \{-1, +1\} \tag{1}$$

where $X_i$ is the symbol from encoder $i$, and $z$ is AWGN with mean 0 and variance $\sigma^2$. In the receiver, a common decoder is to correct errors in received superimposed symbols and to seperate them to each user's data.

Multi-user coding for multiple-access channel(MAC) has been studied for two decades [1]-[8]. For example, the multi-user block code [1] and multi-user convolutional code [2] were proposed for separating the received data from users over noiseless MAC. Multi-user coding for noisy MAC was investigated in [1] [3] [4]. In the noisy

*Dept. of Intelligent Information Eng. & Sci., Doshisha University, Kyotanabe, Kyoto 610-0321 Japan. Email: jcheng@ieee.org, watanabe@muc.doshisha.ac.jp.

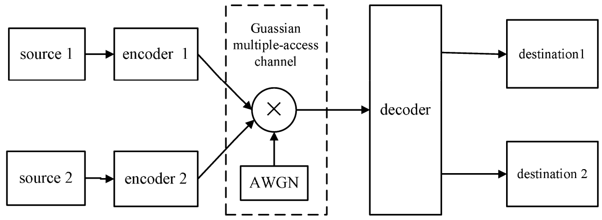



Figure 1: Two-user multiple-access system over GMAC.

MAC, due to interference between users and noise, the multi-user coding with high error-correcting capability is required. It is well-know that an iterative decoding has a potential to give a high cabability to correct errors. In [5], a turbo block code for two users was reported. However, the error correction and users' data separation were carried independently in [5], where the channel error was first corrected by iterative decoding, and then the received data was separated into user' one. Because of this independent processing, a good performance can not provided.

On the other hand, as a single-user correcting error code, Li et al. reported a turbo-Hadamard code (THC) near to the limit of Shannon [9] [10]. The main feature of THC is that the output of its encoder consists of the rows of Hadamard matrix, which is to enhance error-correcting capability. In addition, Li et al. reported an application of the THC to multi-user in [6], where users' data are separated based on interleaves.

In this paper, we investigate a two-user THC for synchroniged GMAC. While the interleaves is used to separate users' data in [6], we here employ the orthogornality of Hadamard. User $i$ ($i = 1, 2$) has his own encoder $C_i$ based on THC, and is assigned a sub-Hadamard matrix. The encoder $C_i$ consists of two component encoders in parallel concatenation. The output of component encoders is rows of the sub-Hadamard matrix. Employing different sub-Hadamard matrices guarantees the orthogornality of the two outputs of users' encoders. In the receiver, each user has its own trellis stage in its component decoders. The orthogonality between users' sub-Hadamard matrices assures that user's component decoder with its own trellis stage gives the posterior probabilities of user's symbols and thus separate users' data.

ISBN: 978-988-17012-2-0

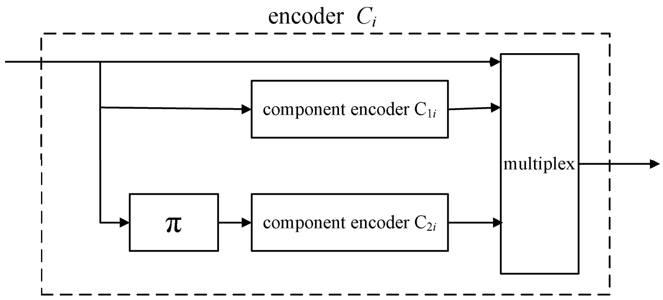



Figure 2: Block diagram of the turbo encoder $C_i$ ($i = 1, 2$) with interleaver $\pi$.

The iterative decoding between the component decoders is to recover the received data from noise.

At this point, we must emphasize that our procedure is quite different from the concept of the conventional works, such as coded CDMA system. In a conventional coded CDMA system, the two basic issues, channel noise and mutual interference between users, are considered separately and independently. In fact, data from a user is first encoded by a forward error control code, and a coded bit is then spread by a spreading code. The forward error control code combats channel noise by adding redundancy, while the spreading code for CDMA focuses mostly on user interference by a cross-correlation to maintain the reliability of communication. However, we follow a procedure designed to simultaneously solve these two conflicting issues by using a two-user coding to cancel both channel noise and mutual interference.

In the next section, we describe two-user coding based on THC. Section 3 gives a decoding procedure. In Section 4, we evaluate the BER performance. Finally in Section 5, we conclude this paper with discussion.

## 2 Two-user coding based on THC

Two-user coding is to assign two constitute codes to two users who can communicate simultaneously with a common receiver through the GMAC. The two-user coding is to deal with both channel noise and mutual interference between users. For the purpose of separating the superimposed symbols from users, it is hoped that the output of one user's encoder is orthogonal to that of the other user's encoder such that the superimposed symbols can be separated.

We employ a turbo encoder $C_i$ for user $i$ ($i = 1, 2$). In encoder $C_i$, two component encoders $C_{1i}$ and $C_{2i}$ are parallel concatenated through an interleaver $\pi$ as shown in Fig. 2. The component encoder $C_{ji}$ has parity check, recursive systematic convolutional (RSC) encoder and sub-Hadamard encoder $\mathcal{H}_{ni}$ (see Fig. 3). The coding procedure in the component encoder $C_{ji}$ is as follows.

1) The information bit stream of user $i(= 1, 2)$ is segmented into blocks $\boldsymbol{d}_{ki}$, $k = 1, 2, \ldots, K$. Each block $\boldsymbol{d}_{ki}$ contains $N = rK$ bits.

2) Caculate the bit $q'_{ki}$, which is the parity check of vector $\boldsymbol{d}_{ki}$.

3) The parity check bit $q'_{ki}$ is input to RSC encoder having a delay element $D$ with a coding rate of 1/2. The output of RSC encoder is a parity check bit $q_{ki}$.

Up to now, the encoding procedure in the component encoder is same as the single-user THC in [9]. While all the rows in Hadamard matrix $H_n$ are used for encoding $q_{ki}$, $\boldsymbol{d}_{ki}$ in [9], we here use a sub-Hadamard matrix. The sub-Hadamard encoder $\mathcal{H}_{ni}$ encodes them by the sub-Hadamard matrix $H_{ni}$. The matrix $H_{n1}$ for user 1 is the first half of Hadamard matrix, while the matrix $H_{n2}$ for user 2 is the second half of Hadamard matrix. Thus the output of component encoder $C_{j1}$ is orthogonal to that of $C_{j2}$. For example, let $n = 4$. The first half matrix $H_{41}$, for user 1, consists of the 1st and 2nd rows of Hadamard matrix $H_4$. The remaining rows contribute to the second half matrix $H_{42}$ for user 2 (see Fig. 4).

The sub-Hadamard encoder $\mathcal{H}_{ni}$ processes as follow.

4) Inputs to $\mathcal{H}_{ni}$ are $q_{ki}$, $\boldsymbol{d}_{ki}$. The encoder $\mathcal{H}_{ni}$ choose a row from the sub-Hadamard matrices $\pm H_{ni}$ as an output. In the row, the first bit is $q_{ki}$, and the 2nd, 3rd, ..., $(2^{r-1} + 1)$-th bits are $\boldsymbol{d}_{ki}$.

The encoding procedure in the component encoders $C_{ji}$ above shows that the output of $C_{ji}$ is the rows of sub-Hadamard matrices $\pm H_{ni}$. Since the rows in $\pm H_{n1}$ are orthogonal to those in $\pm H_{n2}$, it is confirmed that the output of the encoder $C_1$ is orthogonal to that of $C_2$. This make is possible to separate the data from user in the receiver.

## 3 Decoding

In receiver, the two-user decoder is to correct errors caused by channel noise and to separate the received superimposed data to users' data.

The decoder, shown in Fig. 5, consists of two soft input/output component decoders (decoders $D_1$ and $D_2$) with interleaver $\pi$ and deinterleaver $\pi^{-1}$. According to iterative decoding algorithm, the decoder $D_1$ produces an estimate of the extrinsic probabilities of received symbols from the received data. The extrinsic probabilities are then passed through the interleaver to $D_2$ as a prior probabilities. Similarly, $D_2$ calculates its extrinsic probabilities and passes them back through the deinterleaver to $D_1$. After some number of iterations, the decoder converges to an estimate of received symbols.

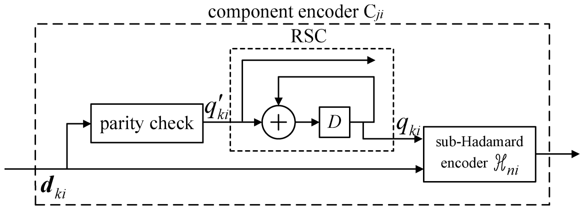



Figure 3: Component encoder with the generation polynomial $G(D) = D/(1 + D)$ in RSC.

$$H_4 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ \hdashline 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} = \begin{bmatrix} H_{41} \\ \hdashline H_{42} \end{bmatrix}$$

$$-H_4 = \begin{bmatrix} -1 & -1 & -1 & -1 \\ -1 & 1 & -1 & 1 \\ \hdashline -1 & -1 & 1 & 1 \\ -1 & 1 & 1 & -1 \end{bmatrix} = \begin{bmatrix} -H_{41} \\ \hdashline -H_{42} \end{bmatrix}$$

Figure 4: Sub-Hadamard matrices $\pm H_{4i}$ for user $i$ ($i = 1, 2$).

In our previous work [8], two-user *four*-states trellis stage in $D_1(D_2)$ was used by the BCJR algorithm to calculate the posterior probabilities of the received symbols and thus their extrinsic probabilities. In this paper, we divide $D_1(D_2)$ into two component decoders, one for user 1 and the other for user 2. We will see that the component decoder has a *two*-state trellis stage.

Before proceeding, let observe the received symbols to GMAC in sender. Due to superposition of the symbols from two users, the received symbols are $u_j = \pm 2, 0$, $(j = 1, 2, \ldots, N)$ under the assumption that there does not exist noise in the channel. The symbol $u_j = \pm 2$ is obviously superimposed from user 1's symbol $u_{j1} = \pm 1$ and user 2's symbol $u_{j2} = \pm 1$. However, for the received symbol $u_j = 0$, it is ambiguous since there are two cases, a) $u_{j1} = +1$ and $u_{j2} = -1$, and b) $u_{j1} = -1$ and $u_{j2} = +1$. For the purpose of discrimination, we denote $u_j = 0_1$ for the former case, and $u_j = 0_2$ for later case. This is shown in Table 1.

Let $\boldsymbol{y}$ be the received data input to the decoder $D_1$ ($D_2$). Then the posterior probabilities $P(u_j|\boldsymbol{y})$ of the received symbols can be seen a combination of the posterior probabilities $P(u_{j1}|\boldsymbol{y})$ of first user's symbols and $P(u_{j2}|\boldsymbol{y})$ of second user's symbols, i.e.,

$$P(u_j|\boldsymbol{y}) = P(u_{j1}|\boldsymbol{y})P(u_{j2}|\boldsymbol{y}) \tag{2}$$

with $u_j = u_{j1} + u_{j2}$. Specially,

$$\begin{aligned} P(u_j = +2|\boldsymbol{y}) &= P(u_{j1} = +1|\boldsymbol{y})P(u_{j2} = +1|\boldsymbol{y}) \\ P(u_j = -2|\boldsymbol{y}) &= P(u_{j1} = -1|\boldsymbol{y})P(u_{j2} = -1|\boldsymbol{y}) \end{aligned}$$

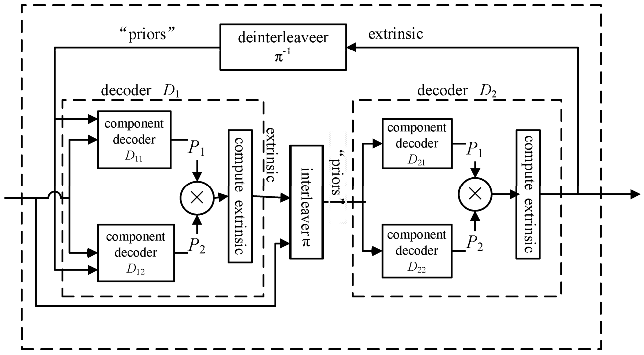



Figure 5: Structure of turbo decoder with interleaver $\pi$ and deinterleaver $\pi^{-1}$.

Table 1: Received symbols $u_j$ is a supuerimpose of users' symbols $u_{j1}$ and $u_{j2}$.

| $u_{j1}$ | $u_{j2}$ | $u_j$ |
|---|---|---|
| $+1$ | $+1$ | $+2$ |
| $-1$ | $-1$ | $-2$ |
| $+1$ | $-1$ | $0_1$ |
| $-1$ | $+1$ | $0_2$ |

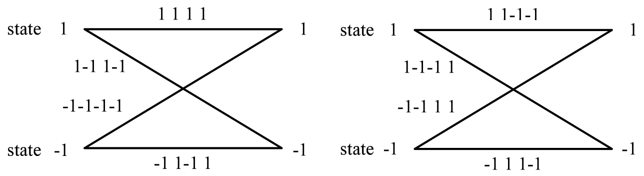
$$\begin{aligned} P(u_j = 0_1|\boldsymbol{y}) &= P(u_{j1} = +1|\boldsymbol{y})P(u_{j2} = -1|\boldsymbol{y}) \\ P(u_j = 0_2|\boldsymbol{y}) &= P(u_{j1} = -1|\boldsymbol{y})P(u_{j2} = +1|\boldsymbol{y}). \end{aligned}$$

We now divide the decoder $D_1$ into two component decoders $D_{11}$ and $D_{12}$. The component decoder $D_{11}$ is to compute $P(u_{j1}|\boldsymbol{y})$ for user 1, and the component decoder $D_{12}$ is to compute $P(u_{j2}|\boldsymbol{y})$ for user 2, both from the received data $\boldsymbol{y}$.

Let the $D_{11}$ be associated with $C_{11}$. Then we have a trellis stage (see Fig. 6(a)) for $D_{11}$ with the labels of branch from the first and second rows of the Hadamard matrices $\pm H_4$ (see Fig. 4). For example, the label of branch that changes from state 1 to state 1 is 1 1 1 1 (Fig. 6(a)), which is the first row of Hadamard matrix $H_4$. Similarly, we have a trellis stage in $D_{12}$ for user 2 with labels from the third and fourth rows of the Hadamard matrices $\pm H_4$ (see Fig. 6(b)). It should be emphasized that labels of branch in $D_{11}$ is orthogonal to those of $D_{12}$, since the rows of Hadamard are orthogonal each other. The orthogonality makes it possible to parallel calculate $P(u_{j1}|\boldsymbol{y})$ in $D_{11}$ and $P(u_{j2}|\boldsymbol{y})$ in $D_{12}$, respectively.

Now we describe how to calculate $P(u_{j1}|\boldsymbol{y})$ and $P(u_{j2}|\boldsymbol{y})$. In the BCJR algorithm in [11], it is well known that a symbol posterior probability is computed by combining $\alpha_t(p)$, $\beta_{t+1}(q)$ and $\gamma_t(p, q)$. Here, $\alpha_t(p)$ represents the probability of the observations up to time $t - 1$, with the state ending in state $p$ at time $t$, $\beta_{t+1}(q)$ represents the

ISBN: 978-988-17012-2-0




(a) Trellis stage (user 1). (b) Trellis stage (user 2).

Figure 6: The trellis stages of users with RSC encoder having the polynomial $G(D) = D/(1 + D)$.

probability of the future observed sequence given that it starts at state $q$ at time $t + 1$, and $\gamma_t(p, q)$ represents the probability of the transtition from state $p$ to state $q$, with the observation at time $t$. We calculate $\gamma_t(p, q)$ from the likelihood probability $p(\boldsymbol{y}|\boldsymbol{h}^{p,q})$. The likelihood probability $p(\boldsymbol{y}|\boldsymbol{h}^{p,q})$ represents the probability of component vector $\boldsymbol{y}$ of the received data conditioned on the transition from state $p$ to state $q$ with the label $\boldsymbol{h}$. For the BCJR algorithm, see [11] for details. Since $\boldsymbol{y}$ is superimposed from the two rows of Hadamard matrices $\pm H_4$, the orthogonality confirms that $D_{11}$ with its trellis stage gives $P(u_{j1}|\boldsymbol{y})$, and $D_{12}$ outputs $P(u_{j2}|\boldsymbol{y})$.

From the posterior probabilities $P(u_{j1}|\boldsymbol{y})$ and $P(u_{j2}|\boldsymbol{y})$ of user's symbols, by (2) we have the posterior probabilities $P(u_j|\boldsymbol{y})$ of received symbols, and thus the log likelihood ratio, i.e.,

$$\Lambda_1(\hat{u_j}) = \ln\frac{P(u_j = +2|\boldsymbol{y})}{P(u_j = -2|\boldsymbol{y})} \quad (3)$$

$$\Lambda_2(\hat{u_j}) = \ln\frac{P(u_j = 0_1|\boldsymbol{y})}{P(u_j = -2|\boldsymbol{y})} \quad (4)$$

$$\Lambda_3(\hat{u_j}) = \ln\frac{P(u_j = 0_2|\boldsymbol{y})}{P(u_j = -2|\boldsymbol{y})} \quad (5)$$

We rewrite above equations as

$$\begin{aligned}\Lambda_1(\hat{u_j}) &= \ln\frac{P(\boldsymbol{y}|u_j = +2)}{P(\boldsymbol{y}|u_j = -2)} + \ln\frac{P(u_j = +2)}{P(u_j = -2)} \\ &= \Lambda_1(\boldsymbol{y}|u_j) + \Lambda_1(u_j) \quad (6)\end{aligned}$$

$$\begin{aligned}\Lambda_2(\hat{u_j}) &= \ln\frac{P(\boldsymbol{y}|u_j = 0_1)}{P(\boldsymbol{y}|u_j = -2)} + \ln\frac{P(u_j = 0_1)}{P(u_j = -2)} \\ &= \Lambda_2(\boldsymbol{y}|u_j) + \Lambda_2(u_j) \quad (7)\end{aligned}$$

$$\begin{aligned}\Lambda_3(\hat{u_j}) &= \ln\frac{P(\boldsymbol{y}|u_j = 0_2)}{P(\boldsymbol{y}|u_j = -2)} + \ln\frac{P(u_j = 0_2)}{P(u_j = -2)} \\ &= \Lambda_3(\boldsymbol{y}|u_j) + \Lambda_3(u_j) \quad (8)\end{aligned}$$

Here $\Lambda_1(u_j)$ ($\Lambda_2(u_j)$, $\Lambda_3(u_j)$) is a prior value about $u_j$ with initial value 0 at initial iteration, and $\Lambda_1(\boldsymbol{y}|u_j)$ ($\Lambda_2(\boldsymbol{y}|u_j)$, $\Lambda_3(\boldsymbol{y}|u_j)$) is a sum of channel value and posterior value. Note that the channel value can be obtained from received symbol $u_j$. Therefore, we have the extrinsic value from the log likehood ratio $\Lambda_1(u_j)$ ($\Lambda_2(u_j)$, $\Lambda_3(u_j)$) by removing the channel value and the prori value.

For the iterative decoding, the extrinsic values obtained in a decoder $D_1$ ($D_2$) are passed through the interleaver to the other decoder $D_2$ ($D_1$) as the prior values $\Lambda_1(\hat{u_j})$, $\Lambda_2(\hat{u_j})$, $\Lambda_3(\hat{u_j})$. After number of iterations, we have the posterior probabilities

$$P(u_j = +2|\boldsymbol{y}) = \frac{\exp(\Lambda_1(\hat{u_j}))}{1 + \exp(\Lambda_1(\hat{u_j})) + \exp(\Lambda_2(\hat{u_j})) + \exp(\Lambda_3(\hat{u_j}))} \quad (9)$$

$$P(u_j = -2|\boldsymbol{y}) = \frac{1}{1 + \exp(\Lambda_1(\hat{u_j})) + \exp(\Lambda_2(\hat{u_j})) + \exp(\Lambda_3(\hat{u_j}))} \quad (10)$$

$$P(u_j = 0_1|\boldsymbol{y}) = \frac{\exp(\Lambda_2(\hat{u_j}))}{1 + \exp(\Lambda_1(\hat{u_j})) + \exp(\Lambda_2(\hat{u_j})) + \exp(\Lambda_3(\hat{u_j}))} \quad (11)$$

$$P(u_j = 0_2|\boldsymbol{y}) = \frac{\exp(\Lambda_3(\hat{u_j}))}{1 + \exp(\Lambda_1(\hat{u_j})) + \exp(\Lambda_2(\hat{u_j})) + \exp(\Lambda_3(\hat{u_j}))} \quad (12)$$

from $\Lambda_1(\hat{u_j})$, $\Lambda_2(\hat{u_j})$, $\Lambda_3(\hat{u_j})$. By choosing the maximum value from (9)-(12), we have the estimates of transmitted symbols $u_j$ ($j = 1, 2, \ldots, N$). Separating the received symbols with Table 1, we obtained users' symbols $u_{j1}$ and $u_{j2}$ ($j = 1, 2, \ldots, N$), thus the users' data.

## 4 Simulation

We evaluate the bit error rate (BER) of the proposed two-user coding by computer simulation, and compare it with the single-user THC. In the two-user coding, each component encoder has a same structure with the RSC encoder having a generation polynomial of $G(D) = D/(1 + D)$. The interleave length in the random interleaver is $N = 999$. Let $r = 2$. The sum-rate of coding is $R = R_1 + R_2 = 1/7 + 1/7$. On the other hand, for single-user THC, the same component encoders is used except for the Hadamard encoder. The coding rate is $R' = 1/3$. The coding rate for two- and single-user coding have to be different because of the different structure of the two coding shames.

After number 10 of iterations, we give the BER curves of the two- and single-user coding shown in Fig. 7. It is obvious that the proposed two-user coding has almost same BER as that of the single-user coding. This arises from the fact that the outputs of users' encoder are orthogonal to each other.

## 5 Conclusion

We addressed the issues of channel noise and mutual interference between users in synchronized GMAC and devised a unified processes for dealing with both of these

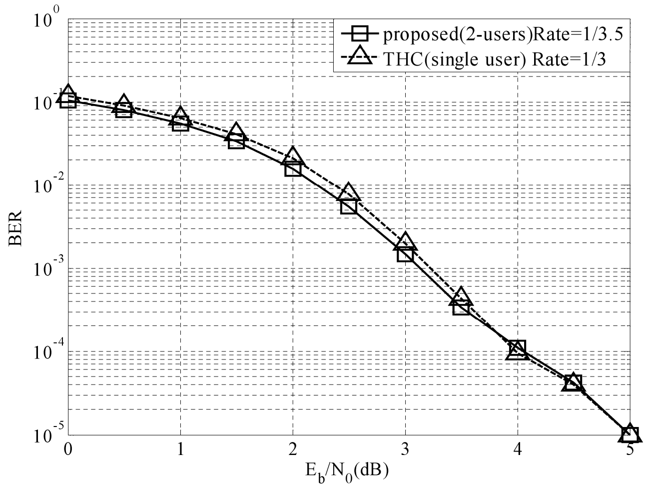



Figure 7: BERs of two- and single-user codes.

issues. We followed a procedure designed to simultaneously solve these two conflicting issues by using a two-user coding. The encoder $C_i$ consists of two component encoders in parallel concatenation. Employing different sub-Hadamard matrices guarantees the orthogornality of the two outputs of users' encoders.

We also proposed a decoding procedure. In the receiver, each user has its own trellis stage in its component decoders. The orthogonality between users' sub-Hadamard matrices assures that user's component decoder with its own trellis stage gives the posterior probabilities of user's symbols and thus separate users' data. The iterative decoding between the component decoders is to recover the received data from noise.

## 6 Acknowledgment

The authors express their thanks to K. Minami for his useful discussions.

## References

[1] S. C. Chang and E. J. Weldon, Jr., "Coding for $T$-user multiple-access channels," *IEEE Trans. Inform. Theory*, vol. IT-25, no. 6, pp. 684–691, Nov. 1979.

[2] R. Peterson and D. J. Costello, "Binary convolutional codes for a multiple-access channel," *IEEE Trans. Inform. Theory*, vol. IT-25, no. 1, pp. 101-105, Jan. 1979.

[3] J. Cheng and Y. Watanabe, "Multiuser $k$-ary code for noisy multiple-access adder channel," *IEEE Trans. Inform. Theory*, vol. IT-47, no. 6, pp. 2603-2607, Sep. 2001.

[4] J. Cheng, T. Ohira, K. Kamoi and Y. Watanabe, "Spreading set with error correction for multiple-access adder channel," *IEEE Trans. Inform. Theory*, vol. IT-57, no. 12, pp. 5524-5529, Dec. 2006.

[5] M.L.M.G. Alcoforado, V.C.da Rocha Jr. and G. Markarian, "Turbo block codes for the binary adder channel," *Proc. Int. Symp. on Inform. Theory* (ISIT), pp. 1883-1887, Sep. 2005.

[6] P. Li, L. Liu, K. Wu and W. K. Leung, "Interleave division multiple-access," *IEEE Trans. Wireless. Commun*, vol. 5, no. 4, pp. 938-947, Apr. 2006.

[7] P. R. Chevillat, "$N$-user trellis coding for a class of multiple-access channels," *IEEE Trans. Inform. Theory*, vol. IT-27, no. 1, pp. 114-120, Jan. 1981.

[8] K. Minami, J. Cheng and Y. Watanabe, "Multiuser turbo-Hadamard code for synchronized Gaussian multiple-access channel," *Proc. 29th Symp. on Inform. Theory and its Applications* (SITA), pp. 819-822, Dec. 2006.

[9] P. Li, W. K. Leung, and K. Y. Wu, "Low-rate turbo-Hadamard codes," *IEEE Trans. Inform. Theory*, vol. 49, no. 12, pp. 3213-3224, Dec. 2003.

[10] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: turbo codes," *Proc. IEEE Int. Communications Conf.* (ICC), vol. 2, pp. 1064-1070, May 1993.

[11] T. K. Moon, Error Correction Coding Mathematical Methods and Algorithms, Utah State University Press: New Jersey, 2005.