PAPER Special Section on Information Theory and Its Applications

A Generalization of the Parallel Error Correcting Codes by Allowing Some Random Errors*

Hideki YAGI[†a)], **Toshiyasu MATSUSHIMA**[††], ***Members***, ***and*** **Shigeichi HIRASAWA**[†††], ***Fellow***

SUMMARY This paper generalizes parallel error correcting codes proposed by Ahlswede et al. over a new type of multiple access channel called parallel error channel. The generalized parallel error correcting codes can handle with more errors compared with the original ones. We show construction methods of independent and non-independent parallel error correcting codes and decoding methods. We derive some bounds about the size of respective parallel error correcting codes. The obtained results imply a single parallel error correcting code can be constructed by two or more kinds of error correcting codes with distinct error correcting capabilities.

key words: *multiple access channel, parallel channel, code construction, error-correcting code, linear block code*

1. Introduction

Coding schemes for multiple access channel have been widely discussed. Especially, for a multiple access adder channel, many code constructions have been proposed [2], [3], [6], [7], [12]. In contrast to conventional works, R. Ahlswede et al. have considered a new model of multiple access channel called a **parallel error channel** and discussed coding schemes for this channel [1].

The parallel error channel is a bundle of m lines through which messages are parallelly transmitted. Although this channel is considered as an instance of **parallel channel** [4], [5], [8], when messages are transmitted through the channel, highly correlated errors occur in respective lines. For example, in a parallel port of a computer, messages are transmitted through several lines simultaneously and disturbed by magnetic noise, etc. At that time, messages at a time instance may be almost equally disturbed. Namely, if an error occurs in a line, the probability that an error occurs in its neighbor lines becomes high. Ahlswede et al. have focused on this fact and, for some positive integer t, introduce a concept of t-**parallel error**, which is defined as the same errors with the Hamming weight less than or equal to t in all lines of the channel. They have derived necessary and sufficient conditions of codes correcting any t-parallel error. They have given code constructions of the optimal parallel error correcting codes with the largest size for given a code length and t. Their work gives a large amount of suggestions, however the channel model in [1] is insufficient for practical applications.

In this paper, we generalize the concept of Ahlswede's parallel error channel, by allowing some random errors along with the common errors in all lines. Subsequently, we derive necessary and sufficient conditions of parallel error correcting codes whose line codes are dependent each other. We show a code construction that achieves the maximal size for a given code length and t. Then, we consider linear parallel error correcting codes whose line codes are independent and derive a bound of the maximal achievable rate (pair of dimensions of all line codes [1], [7]). Therefore, following [1], we first focus on the case of $m = 2$, and then we generalize the results to a general m lines case.

The main contribution of this paper is results for a general m lines case. Contrary to the Ahlswede's parallel error channel, the definition of parallel errors for a general m lines case is not straightforward and we can have several options for the definition. In this paper, we divide m lines of the channel into several groups. Let g denote the number of such groups. Even for a fixed m, the codes with the maximum size can vary depending on the parameter g. Then we discuss the average coding rate per a line by varying either m or g.

This paper is organized as follows: in Sect. 2, we describe a new model and some definitions. Next, in Sect. 3, we derive necessary and sufficient conditions for non-linear parallel error correcting codes whose line codes are dependent each other. Then in Sect. 4, we discuss linear independent parallel error correcting codes. In Sect. 5, we generalize the results obtained in Sect. 3 and 4 for a general m lines case. In Sect. 6, we discuss coding rates of parallel error correcting codes compared with the case where we only use conventional random error correcting codes. Finally in Sect. 7, we state the concluding remarks.

2. Model and Definitions

For a prime power q, let $GF(q)$ be a finite field of order q. For any set $\mathcal{A}$, let $|\mathcal{A}|$ express the size of $\mathcal{A}$. Let n be a positive integer. For any linear space $\mathcal{B} \subseteq GF^n(q)$, let $\dim(\mathcal{B})$ be the dimension of $\mathcal{B}$. For any vector $\boldsymbol{b} \in GF^n(q)$, let $w_H(\boldsymbol{e})$ express the Hamming weight of $\boldsymbol{b}$. We denote the

Manuscript received December 21, 2006.
Manuscript revised April 11, 2007.
Final manuscript received May 26, 2007.

[†]The author is with Media Network Center, Waseda University, Tokyo, 169-8005 Japan.
[††]The author is with the School of Fundamental Science and Engineering, Waseda University, Tokyo, 169-8555 Japan.
[†††]The author is with the School of Creative Science and Engineering, Waseda University, Tokyo, 169-8555 Japan.

*Some content of this paper was presented at 2006 IEEE Information Theory Workshop (ITW2006), Oct. 22–26, 2006, Chengdu, China [13].

a) E-mail: yagi@hirasa.mgmt.waseda.ac.jp

DOI: 10.1093/ietfec/e90–a.9.1745

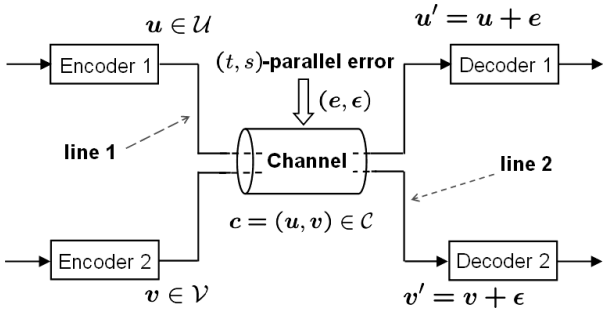

Fig. 1 Illustration of a channel model for $m = 2$ lines case.

support of a vector $\boldsymbol{b} = (b_1, b_2, \ldots, b_n) \in GF^n(q)$ by $\mathcal{S}(\boldsymbol{b}) = \{j | b_j \neq 0\}$. For any sets $\mathcal{A} \subseteq GF^n(q)$ and $\mathcal{B} \subseteq GF^n(q)$, we define the **direct sum** of these sets by $\mathcal{A} + \mathcal{B} = \{\boldsymbol{a} + \boldsymbol{b} | \boldsymbol{a} \in \mathcal{A}, \boldsymbol{b} \in \mathcal{B}\}$.

In this section, we describe a channel model of this paper. In this paper, the two lines model is important to derive results for a general $m(\geq 3)$ lines case. Hence we here describe the two lines model.

Figure 1 shows the channel model considered in this paper. We denote input alphabets of the two lines by $\mathcal{X}$ and $\mathcal{Y}$. In this paper, we assume that the set of the input and output alphabets is a finite field $GF(q)$ for a given prime power q.

Assume that a codeword of a code $C \subset \mathcal{X}^n \times \mathcal{Y}^n$ of length $2n$ is input to the parallel error channel where the first n symbols of the codeword are the first line's message and the last n symbols of it are another line's message. Let $\boldsymbol{e} \in GF^n(q)$ and $\boldsymbol{\epsilon} \in GF^n(q)$ be error vectors in the first line and the second line, respectively. We denote concatenation of these error vectors $\boldsymbol{e}$ and $\boldsymbol{\epsilon}$ by $(\boldsymbol{e}, \boldsymbol{\epsilon}) \in GF^{2n}(q)$. In the channel, there occurs an error vector $(\boldsymbol{e}, \boldsymbol{\epsilon}) \in GF^{2n}(q)$ defined as follows:

Definition 1: Assume that an error vector $(\boldsymbol{e}, \boldsymbol{\epsilon}) \in GF^{2n}(q)$ such that $\boldsymbol{e}, \boldsymbol{\epsilon} \in GF^n(q)$ satisfying

$$w_H(\boldsymbol{e}) \leq t + s, \quad w_H(\boldsymbol{\epsilon}) \leq t + s \tag{1}$$

and

$$w_H(\boldsymbol{e} - \boldsymbol{\epsilon}) \leq 2s \tag{2}$$

occurs and disturbs the input codeword. We call this pair of errors $(\boldsymbol{e}, \boldsymbol{\epsilon})$ a (t, s)-**parallel error**. In this paper, we assume $t \geq s$. □

We can regard that the parallel error $(\boldsymbol{e}, \boldsymbol{\epsilon}) \in GF^{2n}(q)$ consists of two kinds of error symbols: **common error symbols** and **distinct error symbols**. Equations (1) and (2) imply that the number of common error symbols are not greater than t, and the number of distinct error symbols in each line are not greater than s. Figure 2 shows the relationship of these error symbols.

We define codes that can correct any (t, s)-parallel errors.

Definition 2: Let $(\boldsymbol{e}, \boldsymbol{\epsilon})$ and $(\boldsymbol{e}', \boldsymbol{\epsilon}')$ such that $\boldsymbol{e}, \boldsymbol{e}' \in GF^n(q)$ and $\boldsymbol{\epsilon}, \boldsymbol{\epsilon}' \in GF^n(q)$ be two distinct (t, s)-parallel errors. Assume that a code $C \subset \mathcal{X}^n \times \mathcal{Y}^n$ has no pair of distinct codewords $\boldsymbol{c} = (\boldsymbol{u}, \boldsymbol{v}), \boldsymbol{c}' = (\boldsymbol{u}', \boldsymbol{v}') \in C$ such that $\boldsymbol{u}, \boldsymbol{u}' \in \mathcal{X}^n$ and $\boldsymbol{v}, \boldsymbol{v}' \in \mathcal{Y}^n$ satisfying

$$\boldsymbol{c} + (\boldsymbol{e}, \boldsymbol{\epsilon}) = \boldsymbol{c}' + (\boldsymbol{e}', \boldsymbol{\epsilon}'). \tag{3}$$

Then the code C of length n, the number of codewords $|C|$ which can correct any (t, s)-parallel errors is called an $(n, t, s, |C|)$ **parallel error correcting code**, or in short, an $(n, t, s, |C|)$ **P-code** over $GF(q)$. □

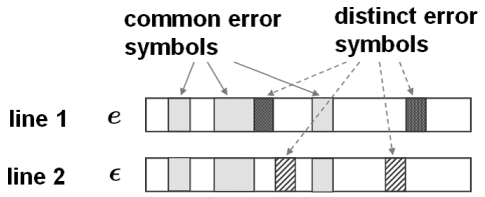

Fig. 2 The relationship of **common error symbols** and **distinct error symbols**, where the numbers of common error symbols and distinct error symbols are at most t and $2s$, respectively.

Definition 3: An $(n, t, s, |C|)$ P-code $C \subset \mathcal{X}^n \times \mathcal{Y}^n$ is called **independent**, or in short, an $(n, t, s, |C|)$ **IP-code** of length n and the number of codewords $|C|$ if the code C is a Cartesian product of subspaces $\mathcal{U} \subseteq \mathcal{X}^n$ and $\mathcal{V} \subseteq \mathcal{Y}^n$, i.e., $C = \mathcal{U} \times \mathcal{V}$. If $\mathcal{U}$ and $\mathcal{V}$ are linear subspaces with $k = \dim(\mathcal{U})$ and $l = \dim(\mathcal{V})$, an $(n, t, s, |C|)$ IP-code C is called **linear**, or in short, an (n, t, s, k, l) **LIP-code** of length n, the dimension of line codes k and l. □

If a code C is an $(n, t, s, |C|)$ IP-code, it consists of two line codes, namely $\mathcal{U}$ and $\mathcal{V}$. We denote codewords of the line codes $\mathcal{U}$ and $\mathcal{V}$ by $\boldsymbol{u}$ and $\boldsymbol{v}$, respectively. In this case, any codeword of C is expressed as $\boldsymbol{c} = (\boldsymbol{u}, \boldsymbol{v}) \in GF^{2n}(q)$. We denote the received sequences corresponding to $\boldsymbol{u}$ and $\boldsymbol{v}$ by $\boldsymbol{u}' = \boldsymbol{u} + \boldsymbol{e}$ and $\boldsymbol{v}' = \boldsymbol{v} + \boldsymbol{\epsilon}$, respectively. Figure 1 illustrates an example of a parallel error channel model for $m = 2$ lines case.

Note that if $s = 0$, a $(t, 0)$-parallel error $(\boldsymbol{e}, \boldsymbol{\epsilon})$ satisfies $\boldsymbol{e} = \boldsymbol{\epsilon}$ and this model is reduced to that assumed in [1]. Therefore, the above model is a generalized version of that in [1] by allowing at most s additional errors in each line of the channel. The definitions of an $(n, t, s, |C|)$ P-code, IP-code and an (n, t, s, k, l) LIP-code are identical to those in [1] when $s = 0$.

Throughout this paper, we denote the maximum size of t-error correcting codes of the length n by $A(n, t)$ and the maximum dimension of linear t-error correcting codes of the length n by $L(n, t)$.

3. Non-independent Parallel Error Correcting Code

In this section, we consider parallel error correcting codes whose line codes are not independent (i.e., $(n, t, s, |C|)$ P-code). The two encoders associated with each line cooperate to generate a codeword $\boldsymbol{c} \in C$.

3.1 Optimum P-Codes

We first derive necessary and sufficient conditions for non-independent parallel error correcting codes.

Let $\mathcal{U} \subseteq \mathcal{X}^n$ and $\mathcal{V} \subseteq \mathcal{Y}^n$. For $\mathcal{U}$ and $\mathcal{V}$, let C_0 be the maximal subspace such that $\mathcal{U} = C_0 + \mathcal{U}_0$ and $\mathcal{V} = C_0 + \mathcal{V}_0$ for some $\mathcal{U}_0 \subseteq \mathcal{X}^n$, $\mathcal{V}_0 \subseteq \mathcal{Y}^n$, i.e., $\mathcal{U} = \{\boldsymbol{u} = \boldsymbol{x} + \boldsymbol{u}_0 | \boldsymbol{x} \in C_0, \boldsymbol{u}_0 \in \mathcal{U}_0\}$ and $\mathcal{V} = \{\boldsymbol{v} = \boldsymbol{x} + \boldsymbol{v}_0 | \boldsymbol{x} \in C_0, \boldsymbol{v}_0 \in \mathcal{V}_0\}$.

We show the following lemma which holds for both non-linear and linear $(n, t, s, |C|)$ P-codes.

Lemma 1: Assume that a code $C \subset \mathcal{X}^n \times \mathcal{Y}^n$ has codewords expressed as

$$\boldsymbol{c} = (\boldsymbol{x} + \boldsymbol{u}_0, \boldsymbol{x} + \boldsymbol{v}_0) \tag{4}$$

where $\boldsymbol{x} + \boldsymbol{u}_0 \in \mathcal{U}, \boldsymbol{x} + \boldsymbol{v}_0 \in \mathcal{V}$ and $\boldsymbol{x} \in C_0$. The code C is an $(n, t, s, |C|)$ P-code if and only if (iff) the following conditions hold:

(i) The subspace C_0 is a $(t + s)$-error correcting code.

(ii) For $\mathcal{U}_0, \mathcal{V}_0$ given by C_0 in the condition (i), define

$$\mathcal{Z} = \{\boldsymbol{u}_0 - \boldsymbol{v}_0 | \boldsymbol{u}_0 \in \mathcal{U}_0, \boldsymbol{v}_0 \in \mathcal{V}_0\}. \tag{5}$$

Then $\mathcal{Z}$ is a $(2s)$-error correcting code of the size $|\mathcal{Z}| = |\mathcal{U}_0| \times |\mathcal{V}_0|$.

(**Proof**) We will prove the if part, assuming that the conditions (i) and (ii) hold.

Let the code C be not an $(n, t, s, |C|)$ P-code. Then from Eq. (3) for $\boldsymbol{c} = (\boldsymbol{x} + \boldsymbol{u}_0, \boldsymbol{x} + \boldsymbol{v}_0), \boldsymbol{c}' = (\boldsymbol{x}' + \boldsymbol{u}_0', \boldsymbol{x}' + \boldsymbol{v}_0') \in C$ such that $\boldsymbol{x}, \boldsymbol{x}' \in C_0$, we have

$$\boldsymbol{x} + \boldsymbol{u}_0 + \boldsymbol{e} = \boldsymbol{x}' + \boldsymbol{u}_0' + \boldsymbol{e}', \tag{6}$$

$$\boldsymbol{x} + \boldsymbol{v}_0 + \boldsymbol{\epsilon} = \boldsymbol{x}' + \boldsymbol{v}_0' + \boldsymbol{\epsilon}' \tag{7}$$

where $(\boldsymbol{e}, \boldsymbol{\epsilon})$ and $(\boldsymbol{e}', \boldsymbol{\epsilon}')$ are (t, s)-parallel errors. Suppose that $\boldsymbol{u}_0 \neq \boldsymbol{u}_0'$ or $\boldsymbol{v}_0 \neq \boldsymbol{v}_0'$. Subtracting Eq. (7) from Eq. (6), we have

$$(\boldsymbol{u}_0 - \boldsymbol{v}_0) - (\boldsymbol{u}_0' - \boldsymbol{v}_0') = (\boldsymbol{e}' - \boldsymbol{\epsilon}') - (\boldsymbol{e} - \boldsymbol{\epsilon}). \tag{8}$$

Let $d_H(\cdot, \cdot)$ denote the Hamming distance. Since

$$\begin{aligned} d_H(\boldsymbol{u}_0 - \boldsymbol{v}_0, \boldsymbol{u}_0' - \boldsymbol{v}_0') &= d_H(\boldsymbol{e}' - \boldsymbol{\epsilon}', \boldsymbol{e} - \boldsymbol{\epsilon}) \\ &\leq w_H(\boldsymbol{e}' - \boldsymbol{\epsilon}') + w_H(\boldsymbol{e} - \boldsymbol{\epsilon}) \leq 4s, \end{aligned} \tag{9}$$

from the definition of Eq. (2), Eq. (8) implies the set $\mathcal{Z}$, given by Eq. (5), is not a $(2s)$-error correcting code (note that the condition $|\mathcal{Z}| = |\mathcal{U}_0| \times |\mathcal{V}_0|$ implies $\boldsymbol{u}_0 - \boldsymbol{v}_0 \neq \boldsymbol{u}_0' - \boldsymbol{v}_0'$ unless $\boldsymbol{u}_0 = \boldsymbol{u}_0'$ and $\boldsymbol{v}_0 = \boldsymbol{v}_0'$). This contradicts the assumption and C is an $(n, t, s, |C|)$ P-code if $\boldsymbol{u}_0 \neq \boldsymbol{u}_0'$ or $\boldsymbol{v}_0 \neq \boldsymbol{v}_0'$.

Next suppose that $\boldsymbol{c} \neq \boldsymbol{c}'$ but $\boldsymbol{u}_0 = \boldsymbol{u}_0'$ and $\boldsymbol{v}_0 = \boldsymbol{v}_0'$. In this case, similar to the proof of Lemma 1 in [1], the subspace C_0 may not be a $(t+s)$-error correcting code. Actually, from Eqs. (6) and (7), we have

$$\boldsymbol{x} - \boldsymbol{x}' = \boldsymbol{e}' - \boldsymbol{e}, \tag{10}$$

$$\boldsymbol{x} - \boldsymbol{x}' = \boldsymbol{\epsilon}' - \boldsymbol{\epsilon} \tag{11}$$

for $\boldsymbol{x}, \boldsymbol{x}' \in C_0$. These equations imply that the subspace C_0 is not a $(t + s)$-error correcting code and this is contradiction to the assumption.

Next, we will prove the only-if part, assuming that the code C is an $(n, t, s, |C|)$ P-code.

Suppose that the condition (ii) does not hold. There exist $\boldsymbol{u}_0, \boldsymbol{u}_0' \in \mathcal{U}_0$ and $\boldsymbol{v}_0, \boldsymbol{v}_0' \in \mathcal{V}_0$ satisfying Eq. (8), and hence Eq. (3) if $\boldsymbol{x} = \boldsymbol{x}'$. This contradicts the assumption that C is a P-code, and therefore the condition (ii) holds.

The claim that the condition (i) holds can also be proved in a similar way to the proof of Lemma 1 in [1]. Suppose that the condition (i) does not hold. Then if $\boldsymbol{u}_0 = \boldsymbol{u}_0'$ and $\boldsymbol{v}_0 = \boldsymbol{v}_0'$, there exist $\boldsymbol{x}, \boldsymbol{x}' \in C_0$ satisfying Eqs. (10) and (11), and hence Eq. (3). This is contradiction, and thus the condition (i) holds. Consequently, the conditions (i) and (ii) hold. □

In Ahlswede's model, only the condition for the subspace C_0 is necessary. On the other hand, the channel model in this paper requires the condition for the difference set $\mathcal{Z}$, given by Eq. (5). The $(n, t, s, |C|)$ P-codes have structure of combining two kinds of ordinary error correcting codes, namely, a $(t + s)$-error correcting code and a $(2s)$-error correcting code.

We show the following theorem about the size of a non-independent P-code C.

Theorem 1: Let C be an (n, t, s, M) P-code with $M = |C|$. Then we have the following statements:

(i) The size M is bounded as

$$M \leq A(n, t + s) \times A(n, 2s). \tag{12}$$

(ii) For $M = A(n, t + s) \times A(n, 2s)$, there exists an (n, t, s, M) P-code.

(**Proof**) We will briefly show the statement (i). Apparently, we have $|C_0| \leq A(n, t + s)$ and $|\mathcal{Z}| = |\mathcal{U}_0| \times |\mathcal{V}_0| \leq A(n, 2s)$ from the conditions of Lemma 1. Then $M = |C_0| \times |\mathcal{U}_0| \times |\mathcal{V}_0| \leq A(n, t + s) \times A(n, 2s)$.

Next, we will show that we can construct an (n, t, s, M) P-code which satisfies (ii).

Construction I: Choose any $(t + s)$-error correcting code of the size $A(n, t+s)$ as C_0. We also choose a $(2s)$-error correcting code of the size $A(n, 2s)$ as $\mathcal{V}_0$ and let $\mathcal{U}_0 = \{\boldsymbol{0}\}$, i.e., $\mathcal{U} = C_0$ since $\mathcal{U} = C_0 + \mathcal{U}_0$ from the definition. We define $\mathcal{V}(\boldsymbol{x}) = \{\boldsymbol{x} + \boldsymbol{v}_0 | \boldsymbol{v}_0 \in \mathcal{V}_0\}$ for $\boldsymbol{x} \in C_0$. Then the code is constructed by $C = \{(\boldsymbol{u}, \boldsymbol{v}) | \boldsymbol{u} \in \mathcal{U}, \boldsymbol{v} \in \mathcal{V}(\boldsymbol{u})\}$.

For $\boldsymbol{c} = (\boldsymbol{u}, \boldsymbol{v}), \boldsymbol{c}' = (\boldsymbol{u}', \boldsymbol{v}') \in C$, equations

$$\boldsymbol{u} + \boldsymbol{e} = \boldsymbol{u}' + \boldsymbol{e}', \tag{13}$$

$$\boldsymbol{v} + \boldsymbol{\epsilon} = \boldsymbol{v}' + \boldsymbol{\epsilon}' \tag{14}$$

never hold simultaneously since Eq. (13) for $\boldsymbol{u} \neq \boldsymbol{u}'$ itself implies $\mathcal{U}(= C_0)$ is not a $(t + s)$-error correcting code and Eqs. (13) and (14) for $\boldsymbol{u} = \boldsymbol{u}'$ lead to $\boldsymbol{v}_0 - \boldsymbol{v}_0' = (\boldsymbol{\epsilon}' - \boldsymbol{e}') - (\boldsymbol{\epsilon} - \boldsymbol{e})$, which implies that $\mathcal{V}_0$ is not a $(2s)$-error correcting code (note that $\boldsymbol{v} - \boldsymbol{v}' = (\boldsymbol{u} + \boldsymbol{v}_0) - (\boldsymbol{u}' + \boldsymbol{v}_0') = \boldsymbol{v}_0 - \boldsymbol{v}_0'$ if $\boldsymbol{u} = \boldsymbol{u}'$ and

$d_H(\boldsymbol{\epsilon}-\boldsymbol{e}, \boldsymbol{\epsilon}'-\boldsymbol{e}') \leq 4s$). Hence the code C can correct any (t, s)-parallel error.

Obviously, $M = A(n, t+s) \times A(n, 2s)$. Therefore the code C is an (n, t, s, M) P-code. □

3.2 Decoding Algorithm for P-Codes

We here describe a decoding process of the P-codes mentioned in Lemma 1. Assume that a codeword $\boldsymbol{c} = (\boldsymbol{u}, \boldsymbol{v}) \in C$ has been sent and a sequence $\boldsymbol{c}' = \boldsymbol{c} + (\boldsymbol{e}, \boldsymbol{\epsilon})$ is received by the decoder where errors $(\boldsymbol{e}, \boldsymbol{\epsilon})$ are a (t, s)-parallel error. We denote $\boldsymbol{u}' = \boldsymbol{u} + \boldsymbol{e}$ and $\boldsymbol{v}' = \boldsymbol{v} + \boldsymbol{\epsilon}$.

Decoding Algorithm I:

(1) Calculate $\boldsymbol{z} = \boldsymbol{v}' - \boldsymbol{u}'$.
(2) For $\boldsymbol{z}$, perform a decoding algorithm of the code $\mathcal{Z}$ to find codewords $\boldsymbol{u}_0$ and $\boldsymbol{v}_0$, and an error pattern $\boldsymbol{f} = \boldsymbol{\epsilon} - \boldsymbol{e}$.
(3) Perform a decoding algorithm of the code C_0 by erasing symbols of $\boldsymbol{u}' - \boldsymbol{u}_0$ in the positions of $\mathcal{S}(\boldsymbol{f})$.

We will show that Decoding Algorithm I finds the transmitted codeword $\boldsymbol{c} = (\boldsymbol{u}, \boldsymbol{v}) \in C$ if there occurs a (t, s)-parallel error.

Since $\boldsymbol{u} = \boldsymbol{x} + \boldsymbol{u}_0$ and $\boldsymbol{v} = \boldsymbol{x} + \boldsymbol{v}_0$, we obtain $\boldsymbol{z} = \boldsymbol{v}' - \boldsymbol{u}' = \boldsymbol{v}_0 - \boldsymbol{u}_0 + \boldsymbol{\epsilon} - \boldsymbol{e} = \boldsymbol{v}_0 - \boldsymbol{u}_0 + \boldsymbol{f}$ in the step (1). Since $w_H(\boldsymbol{f}) \leq 2s$ and the code $\mathcal{Z}$ is a $(2s)$-error correcting code, a conventional decoding algorithm for the code $\mathcal{Z}$ can correctly find codewords $\boldsymbol{u}_0$ and $\boldsymbol{v}_0$ from $\boldsymbol{z} = (\boldsymbol{v}_0 - \boldsymbol{u}_0) + \boldsymbol{f}$ (note that the condition $|\mathcal{Z}| = |\mathcal{U}_0| \times |\mathcal{V}_0|$ in Lemma 1 implies there is one-to-one correspondence between $(\boldsymbol{v}_0 - \boldsymbol{u}_0)$ and a pair $(\boldsymbol{u}_0, \boldsymbol{v}_0)$). Then we can obtain the error pattern $\boldsymbol{f}$ by calculating $\boldsymbol{f} = \boldsymbol{z} - (\boldsymbol{v}_0 - \boldsymbol{u}_0)$ in the step (2). In the step (3), we regard symbols of the sequence $\boldsymbol{u}' - \boldsymbol{u}_0$ in the positions of $\mathcal{S}(\boldsymbol{f})$ as erasure symbols. We denote the resultant sequence by $\tilde{\boldsymbol{u}}$. Since the code C_0 is a $(t+s)$-error correcting code, it has a minimum distance $d(C_0) \geq 2(t+s)+1$ and corrects t errors and $2s$ erasure symbols [9], [10]. Therefore we can obtain the codeword $\boldsymbol{x} \in C_0$ from $\tilde{\boldsymbol{u}}$ and subsequently, $\boldsymbol{u} = \boldsymbol{x} + \boldsymbol{u}_0$ and $\boldsymbol{v} = \boldsymbol{x} + \boldsymbol{v}_0$. Thus Decoding Algorithm I surely finds $\boldsymbol{c} = (\boldsymbol{u}, \boldsymbol{v})$.

4. Linear Independent Parallel Error Correcting Code

4.1 Necessary and Sufficient Conditions for LIP-Codes

In this section, we discuss $(n, t, s, |C|)$ IP-codes $C = \mathcal{U} \times \mathcal{V}$. We only consider linear codes as $\mathcal{U} \subseteq \mathcal{X}^n$ and $\mathcal{V} \subseteq \mathcal{Y}^n$, i.e., the code C is an LIP-code.

Lemma 2: For two linear subspaces $\mathcal{U} \subseteq \mathcal{X}^n$ and $\mathcal{V} \subseteq \mathcal{Y}^n$, a code $C = \mathcal{U} \times \mathcal{V}$ is an (n, t, s, k, l) LIP-code with $k = \dim(\mathcal{U})$ and $l = \dim(\mathcal{V})$ iff the following conditions hold:

(i) Let $C_0 = \mathcal{U} \cap \mathcal{V}$. Then C_0 is a linear $(t+s)$-error correcting code.
(ii) The direct sum $\mathcal{U}+\mathcal{V}$ is a linear $(2s)$-error correcting code.

(**Proof**) First we will prove the if part, assuming that the conditions (i) and (ii) hold but the code C is not an LIP-code. There exist $\boldsymbol{c} = (\boldsymbol{u}, \boldsymbol{v}), \boldsymbol{c}' = (\boldsymbol{u}', \boldsymbol{v}') \in C$ and (t, s)-parallel errors $(\boldsymbol{e}, \boldsymbol{\epsilon})$ and $(\boldsymbol{e}', \boldsymbol{\epsilon}')$ which satisfy Eqs. (13) and (14). First suppose $(\boldsymbol{u} - \boldsymbol{u}') \notin C_0$ or $(\boldsymbol{v} - \boldsymbol{v}') \notin C_0$, then it can be easily shown that $\boldsymbol{u} - \boldsymbol{v} \neq \boldsymbol{u}' - \boldsymbol{v}'$. Then from Eqs. (13) and (14), $\boldsymbol{u} - \boldsymbol{v} - (\boldsymbol{u}' - \boldsymbol{v}') = \boldsymbol{f} - \boldsymbol{f}'$ where $\boldsymbol{f} = \boldsymbol{\epsilon} - \boldsymbol{e}$ and $\boldsymbol{f}' = \boldsymbol{\epsilon}' - \boldsymbol{e}'$. Since $w_H(\boldsymbol{f}) \leq 2s, w_H(\boldsymbol{f}') \leq 2s$ and $\boldsymbol{u} - \boldsymbol{v} \in \mathcal{U}+\mathcal{V}, \boldsymbol{u}' - \boldsymbol{v}' \in \mathcal{U}+\mathcal{V}$, this contradicts the assumption that the condition (ii) holds.

The claim that the code C_0 is a $(t+s)$-error correcting code can be proved in a similar way to the proof of Lemma 2 in [1]. Hence we omit the proof here.

As for the only-if part, we can show the claim by assuming the condition (i) or (ii) does not hold. □

Although we cannot obtain the optimal IP-code with the maximum size, considering LIP-codes, we can obtain bounds on achievable rates (pair of dimensions of all line codes [1], [7]) of LIP-codes.

Theorem 2: For given positive integers n, t, s and k_1, k_2, if there exists an (n, t, s, k_1, k_2) LIP-code, k_1 and k_2 satisfy

$$k_1 + k_2 \leq L(n, t+s) + L(n, 2s) \tag{15}$$

and $k_1 \leq n$, $k_2 \leq n$.

(**Proof**) Assume that there exists an (n, t, s, k_1, k_2) LIP-code such that

$$k_1 + k_2 > L(n, t+s) + L(n, 2s). \tag{16}$$

Since $C_0 = \mathcal{U} \cap \mathcal{V}$ from Lemma 2, $\dim(\mathcal{U}+\mathcal{V}) = k_1 + k_2 - \dim(C_0)$. The condition (i) of Lemma 2 requires the subcode C_0 to be a $(t+s)$-error correcting code, whose dimension satisfies $\dim(C_0) \leq L(n, t+s)$. From Eq. (16), we have

$$\begin{aligned}\dim(\mathcal{U}+\mathcal{V}) &\geq k_1 + k_2 - L(n, t+s) \\ &> L(n, 2s),\end{aligned} \tag{17}$$

and thus $\mathcal{U}+\mathcal{V}$ cannot be a $(2s)$-error correcting code. From Lemma 2, the code C cannot be an (n, t, s, k_1, k_2) LIP-code and this contradicts the assumption. □

From Theorem 2, the dimension pair of any (n, t, s, k_1, k_2) LIP-code is upper-bounded by $L(n, t+s) + L(n, 2s)$. The dimension of an LIP-code depends on those of line codes $\mathcal{U}$ and $\mathcal{V}$. Although it is not straightforward to construct an (n, t, s, k_1, k_2) LIP-code with the maximum achievable rate, we can derive some lower-bound on it. Let C' be a linear t-error correcting code with the maximum dimension $L(n, t)$. For some integer $t' > t$, let $K(n, t, t')$ denote the maximum dimension of a linear t'-error correcting code, which is a subcode of C' with the dimension $L(n, t)$.

Theorem 3: For given positive integers n, t, s, if there exist any (n, t, s, k_1, k_2) LIP-codes, we can construct an (n, t, s, k_1, k_2) LIP-code C which satisfies

$$L(n, 2s) + K(n, 2s, t+s) \leq k_1 + k_2 \tag{18}$$

and $k_1 \leq n, k_2 \leq n$.

(**Proof**) The following construction gives an (n, t, s, k_1, k_2) LIP-code which satisfies Eq. (18).

Construction II: We choose any linear $(2s)$-error correcting code such that its dimension is $L(n, 2s)$ and its linear subcode correcting any $t + s$ random errors has the dimension $K(n, 2s, t + s)$ as the code $\mathcal{U} + \mathcal{V}$. We denote $k = K(n, 2s, t + s)$ bases of the linear subcode by $\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, \ldots, \boldsymbol{\alpha}_k$. Letting $k' = L(n, 2s)$, we denote other $k' - k$ bases of $\mathcal{U} + \mathcal{V}$ by $\boldsymbol{\beta}_1, \boldsymbol{\beta}_2, \ldots, \boldsymbol{\beta}_{k'-k}$.

Now we divide $\{1, 2, \ldots, k'-k\}$ into two disjoint sets $\mathcal{I}_1$ and $\mathcal{I}_2$ (with $\mathcal{I}_1 \cap \mathcal{I}_2 = \emptyset$) such that $\{\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, \ldots, \boldsymbol{\alpha}_k\} \cup \{\beta_i | \, i \in \mathcal{I}_1\}$ are bases of $\mathcal{U}$ and $\{\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, \ldots, \boldsymbol{\alpha}_k\} \cup \{\beta_i | \, i \in \mathcal{I}_2\}$ are bases of $\mathcal{V}$. Let $C = \mathcal{U} \times \mathcal{V}$. Then we have $(k_1 - k) + (k_2 - k) = k' - k$ where $k_1 = \dim(\mathcal{U})$ and $k_2 = \dim(\mathcal{V})$. From Lemma 2, the code C is an (n, t, s, k_1, k_2) LIP-code. It is readily seen that the dimension pair (k_1, k_2) satisfies $k_1 + k_2 = L(n, 2s) + K(n, 2s, t + s)$. □

If maximum distance separable (MDS) codes [9], [10] exist for a number of symbols q and a code length n, we can construct an (n, t, s, k_1, k_2) LIP-code which achieves the maximum achievable rate.

Corollary 1: If there exists a q-ary MDS code of a length n, we can construct an (n, t, s, k_1, k_2) LIP-code such that

$$k_1 + k_2 = L(n, t + s) + L(n, 2s) \tag{19}$$

and $k_1 \leq n, k_2 \leq n$.

(**Proof**) From Lemma 2, we choose any linear $(t+s)$-error correcting code of the dimension $k = L(n, t + s)$ as the code $C_0 = \mathcal{U} \cap \mathcal{V}$. Note that such code should be an MDS code in this case since the dimension of a $(t + s)$-error correcting code $\tilde{C}$ is bounded by the well-known Singleton bound $\dim(\tilde{C}) \leq n - d(\tilde{C}) + 1$ where $d(\tilde{C})$ denotes the minimum distance of $\tilde{C}$, and MDS codes satisfy this with equality. Furthermore, we choose a linear $(2s)$-error correcting code C' (which should be also an MDS code) of the dimension $k' = \dim(C') = L(n, 2s)$ which is a super code of C_0. Note that we can always choose such pair of MDS codes since an MDS code of a higher dimension includes an MDS code of a lower dimension as its subspace. In the sequel, we can take the same procedure as Construction II. From Lemma 2, it is obvious the code C is an (n, t, s, k_1, k_2) LIP-code and $k_1 + k_2 = k + k' = L(n, t + s) + L(n, 2s)$. □

For a prime power q, if a positive integer n satisfies $|q - n| \leq 1$, there exist (lengthened) Reed-Solomon codes with any dimension $k \leq n$ [9]. Thus in this case, we can construct an (n, t, s, k_1, k_2) LIP-code which achieves Eq. (19).

4.2 Decoding Algorithm for LIP-Codes

We here mention a decoding process of the LIP-code obtained by Construction II. Let $\mathcal{U}_0$ and $\mathcal{V}_0$ satisfy $\mathcal{U} = C_0 + \mathcal{U}_0$ and $\mathcal{V} = C_0 + \mathcal{V}_0$, respectively. Note that since C_0 is a common linear subspace of $\mathcal{U}$ and $\mathcal{V}$, we can always choose such $\mathcal{U}_0$ and $\mathcal{V}_0$. As in Sect. 3, we assume that a codeword $\boldsymbol{c} = (\boldsymbol{u}, \boldsymbol{v}) = (\boldsymbol{x} + \boldsymbol{u}_0, \boldsymbol{y} + \boldsymbol{v}_0) \in C$ with $\boldsymbol{x}, \boldsymbol{y} \in C_0, \boldsymbol{u}_0 \in \mathcal{U}_0$ and $\boldsymbol{v}_0 \in \mathcal{V}_0$ has been transmitted and a sequence $\boldsymbol{c}' = \boldsymbol{c} + (\boldsymbol{e}, \boldsymbol{\epsilon})$ is received by the decoder where $(\boldsymbol{e}, \boldsymbol{\epsilon})$ is a (t, s)-parallel error. We denote $\boldsymbol{u}' = \boldsymbol{u} + \boldsymbol{e}$ and $\boldsymbol{v}' = \boldsymbol{v} + \boldsymbol{\epsilon}$.

For an LIP-code C, we denote a generator matrix of the code C_0 by G_0. Similarly, we denote a generator matrix of $\mathcal{U}_0$ and $\mathcal{V}_0$ by G_1 and G_2, respectively. The sizes of G_0, G_1, G_2 are $k \times n, (k_1 - k) \times n, (k_2 - k) \times n$, respectively. Let an overall generator matrix of $\mathcal{U} + \mathcal{V}$ be

$$G = \begin{pmatrix} G_0 \\ G_1 \\ G_2 \end{pmatrix} \tag{20}$$

of the size $(k_1 + k_2 - k) \times n$, and then the rank of G is full.

Decoding Algorithm II:

(1) Calculate $\boldsymbol{z} = \boldsymbol{v}' - \boldsymbol{u}'$.
(2) For $\boldsymbol{z}$, perform a decoding algorithm for the code $\mathcal{U}+\mathcal{V}$ to find a codeword $\boldsymbol{v} - \boldsymbol{u}$ and an error pattern $\boldsymbol{f} = \boldsymbol{\epsilon} - \boldsymbol{e}$.
(3) Calculate

$$\boldsymbol{a} = (a_1, a_2, \ldots, a_{k_1+k_2-k}) = (\boldsymbol{v} - \boldsymbol{u})G^{\dagger} \tag{21}$$

where $G^{\dagger} = G^T(GG^T)^{-1}$ is a generalized inverse matrix* (Moore-Penrose pseudo-inverse matrix [11]) of G and calculate $\boldsymbol{u}_0 = (a_{k+1}, \ldots, a_{k_1-k})G_1 \in \mathcal{U}_0$ and $\boldsymbol{v}_0 = (a_{k_1-k+1}, \ldots, a_{k_1+k_2-k})G_2 \in \mathcal{V}_0$.
(4) Calculate $\boldsymbol{u}' - \boldsymbol{u}_0$ and perform a decoding algorithm for the code C_0 by erasing symbols of $\boldsymbol{u}' - \boldsymbol{u}_0$ in the support $\mathcal{S}(\boldsymbol{f})$.
(5) Calculate $\boldsymbol{v}' - \boldsymbol{v}_0$ and perform a decoding algorithm for the code C_0 by erasing symbols of $\boldsymbol{v}' - \boldsymbol{v}_0$ in the support $\mathcal{S}(\boldsymbol{f})$.

We will show the validity of Decoding Algorithm II that it corrects any (t, s)-parallel error.

Note that $\boldsymbol{z} = \boldsymbol{v}' - \boldsymbol{u}' = \boldsymbol{v} - \boldsymbol{u} + \boldsymbol{f}$ in Step (1) and the equation

$$\boldsymbol{v} - \boldsymbol{u} = (\boldsymbol{y} - \boldsymbol{x}) + \boldsymbol{v}_0 - \boldsymbol{u}_0 = \boldsymbol{a}G \tag{22}$$

holds for some $\boldsymbol{a} \in GF^{k_1+k_2-k}(q)$. Since $w_H(\boldsymbol{f}) \leq 2s$, the decoding algorithm for the code $\mathcal{U} + \mathcal{V}$ finds $\boldsymbol{v} - \boldsymbol{u}$ and $\boldsymbol{f}$ in Step (2). If we multiply the generalized inverse matrix $G^{\dagger}$ to each term of Eq. (22) by right,

$$(\boldsymbol{v} - \boldsymbol{u})G^{\dagger} = \boldsymbol{a}GG^{\dagger} = \boldsymbol{a} \tag{23}$$

where the last equation can be obtained by the definition of $G^{\dagger}$ as $GG^{\dagger} = I$ (I denotes the identity matrix). Therefore, in Step (3), we can obtain $\boldsymbol{a}$ and re-encoding operation generates $\boldsymbol{u}_0 \in \mathcal{U}_0$ and $\boldsymbol{v}_0 \in \mathcal{V}_0$. In Step (4), we calculate $\boldsymbol{u}' - \boldsymbol{u}_0 = \boldsymbol{x} + \boldsymbol{e}$ with $\boldsymbol{x} \in C_0$. Since the code C_0 has the minimum distance $d(C_0) \geq 2(t + s) + 1$, this can correct t errors

*The symbol T denotes transposition of a matrix.

and $2s$ erasure symbols [9], [10]. Then from $\boldsymbol{u}' - \boldsymbol{u}_0$, we can obtain $\boldsymbol{x}$ correctly by regarding the symbols of $\boldsymbol{u}' - \boldsymbol{u}_0$ in the support $\mathcal{S}(\boldsymbol{f})$ as erasure symbols. We can show similarly for Step (5) that we can obtain $\boldsymbol{y} \in C_0$ correctly by regarding the symbols of $\boldsymbol{v}' - \boldsymbol{v}_0$ in $\mathcal{S}(\boldsymbol{f})$ as erasure symbols. Consequently, we can correct the (t, s)-parallel error.

5. General Case with $m \geq 3$ Lines

In this section, we generalize the results in Sect. 3 and 4 for $m \geq 3$ lines case.

5.1 Definition of (t, s)-Parallel Error for General Case

First, we define (t, s)-parallel error for $m \geq 3$ lines case. Intuitively, in the Ahlswede's channel model, the definition of $(t, 0)$-parallel error for $m \geq 3$ lines case is unique, i.e, error patterns of all m lines are exactly the same with the Hamming weight less than or equal to t. On the other hand, in the model of this paper, we can consider several options to define (t, s)-parallel error for the general m lines case. Assume that error sequences are denoted by $\boldsymbol{e}_1, \boldsymbol{e}_2, \ldots, \boldsymbol{e}_m$ where each $\boldsymbol{e}_i \in GF^n(q)$ occurs in the i-th line of the parallel error channel. We denote concatenation of these m error sequences by $(\boldsymbol{e}_1, \boldsymbol{e}_2, \ldots, \boldsymbol{e}_m) \in GF^{mn}(q)$.

Definition 4: [Type I (t, s)-Parallel Error] Assume that an error vector $(\boldsymbol{e}_1, \boldsymbol{e}_2, \ldots, \boldsymbol{e}_m)$ satisfies

$$w_H(\boldsymbol{e}_i) \leq t + s \tag{24}$$

for $i = 1, 2, \ldots, m$ and the inequality

$$w_H(\boldsymbol{e}_i - \boldsymbol{e}_{i+1}) \leq 2s \tag{25}$$

for $i = 1, 2, \ldots, m-1$. Then the error vector $(\boldsymbol{e}_1, \boldsymbol{e}_2, \ldots, \boldsymbol{e}_m)$ is referred to as **Type I (t, s)-parallel error**. □

We will consider another model of (t, s)-parallel errors. We denote the indices set of m channel lines by $\mathcal{L} = \{1, 2, \ldots, m\}$. We assume the set $\mathcal{L}$ is divided into g $(1 \leq g \leq \lfloor m/2 \rfloor)$ disjoint subsets $\mathcal{L}_1, \mathcal{L}_2, \ldots, \mathcal{L}_g$ such that $|\mathcal{L}_j| \geq 2$ for $j = 1, 2, \ldots, g$ and elements of $\mathcal{L}_j$ are positive integers from $\sum_{i=1}^{j-1} |\mathcal{L}_i| + 1$ to $\sum_{i=1}^{j} |\mathcal{L}_i|$. For example, $\mathcal{L}_1 = \{1, 2, \ldots, |\mathcal{L}_1|\}$ and $\mathcal{L}_2 = \{|\mathcal{L}_1| + 1, \ldots, m\}$ if $g = 2$.

It is readily seen that the support of common error symbols between $\boldsymbol{e}_i$ and $\boldsymbol{e}_{i'}$ denoted by $\mathcal{T}(\boldsymbol{e}_i, \boldsymbol{e}_{i'})$ can be expressed as

$$\mathcal{T}(\boldsymbol{e}_i, \boldsymbol{e}_{i'}) = (\mathcal{S}(\boldsymbol{e}_i) \cup \mathcal{S}(\boldsymbol{e}_{i'})) \setminus \mathcal{S}(\boldsymbol{e}_i - \boldsymbol{e}_{i'}). \tag{26}$$

We define another model of (t, s)-parallel errors as follows:

Definition 5: [Type II (t, s)-Parallel Error] Assume that an error vector $(\boldsymbol{e}_1, \boldsymbol{e}_2, \ldots, \boldsymbol{e}_m)$ satisfies Eq. (24) for $i = 1, 2, \ldots, m$ and Eq. (25) for $i = 1, 2, \ldots, m-1$. We further assume that the common error symbols of error vectors in lines indexed by $\mathcal{L}_j$ are exactly the same, i.e., the equation

$$\mathcal{T}(\boldsymbol{e}_i, \boldsymbol{e}_{i+1}) = \mathcal{T}(\boldsymbol{e}_{i+1}, \boldsymbol{e}_{i+2}) \tag{27}$$

holds for any $i, i+1, i+2 \in \mathcal{L}_j$ if $|\mathcal{L}_j| \geq 3$. Then the error vector $(\boldsymbol{e}_1, \boldsymbol{e}_2, \ldots, \boldsymbol{e}_m)$ is referred to as **Type II (t, s)-parallel error**. □

Although the restriction of Type II model seems too strict, if we set $s = 0$, this model is reduced to the model of Ahlswede et al. [1]. Note that if m is even and $g = m/2$, the Type II (t, s)-parallel error is reduced to the Type I (t, s)-parallel error. Therefore if m is even, the parallel error channel of Type II model is a wider class than that of Type I model[†].

Now we define parallel error correcting codes for parallel error channel of both Type I and II (t, s)-parallel errors.

Definition 6: For $m \geq 3$, assume that the code C is a subspace of a Cartesian product of m $GF^n(q)$. If there exists no pair of distinct codewords $\boldsymbol{c} = (\boldsymbol{c}_1, \boldsymbol{c}_2, \ldots, \boldsymbol{c}_m), \boldsymbol{c}' = (\boldsymbol{c}'_1, \boldsymbol{c}'_2, \ldots, \boldsymbol{c}'_m) \in C$ with $\boldsymbol{c}_i, \boldsymbol{c}'_i \in GF^n(q)$ satisfying

$$\boldsymbol{c} + (\boldsymbol{e}_1, \boldsymbol{e}_2, \ldots, \boldsymbol{e}_m) = \boldsymbol{c}' + (\boldsymbol{e}'_1, \boldsymbol{e}'_2, \ldots, \boldsymbol{e}'_m), \tag{28}$$

where $(\boldsymbol{e}_1, \boldsymbol{e}_2, \ldots, \boldsymbol{e}_m)$ and $(\boldsymbol{e}'_1, \boldsymbol{e}'_2, \ldots, \boldsymbol{e}'_m)$ are (t, s)-parallel errors, the code C is called an $(n, m, t, s, |C|)$ P-code. □

An $(n, m, t, s, |C|)$ P-code is called an IP-code if $C = C_1 \times C_2 \times \cdots \times C_m = \prod_{i=1}^{m} C_i$ where $C_i \subseteq GF^n(q)$ is the i-th line code. If each line code C_i is a linear code of the dimension k_i, an IP-code is referred to as an $(n, m, t, s, \{k_1, k_2, \ldots, k_m\})$ LIP-code.

5.2 (t, s)-Parallel Error Correcting Code for Type II Model

As mentioned in Sect. 5.1, the Type II model is a wider class of Type I model and therefore we show only results for Type II model. We consider codes that can correct any Type II (t, s)-parallel errors for a general $m \geq 3$ lines case. In this model, the indices set $\mathcal{L}$ of lines is divided into g disjoint subset $\mathcal{L}_1, \mathcal{L}_2, \ldots, \mathcal{L}_g$ with the size $|\mathcal{L}_i| \geq 2$ for $i = 1, 2, \ldots, g$. Defining $l(j) = \sum_{i=1}^{j-1} |\mathcal{L}_i| + 1$ for $j = 2, 3, \ldots, g$, the first index in each $\mathcal{L}_j$ is denoted by $l(j)$.

Lemma 3: Assume that there are subspaces $C_0 \subseteq GF^n(q)$ and $\mathcal{U}_1, \mathcal{U}_2 \subseteq GF^n(q)$ such that any codewords in the first two lines $(\boldsymbol{c}_1, \boldsymbol{c}_2) \in GF^{2n}(q)$ are expressed as

$$(\boldsymbol{c}_1, \boldsymbol{c}_2) = (\boldsymbol{x} + \boldsymbol{u}_1, \boldsymbol{x} + \boldsymbol{u}_2) \tag{29}$$

where $\boldsymbol{x} \in C_0$ and $\boldsymbol{u}_i \in \mathcal{U}_i$. A code C, given by $C = \{(\boldsymbol{c}_1, \boldsymbol{c}_2)\} \times \prod_{i=3}^{m} C_i$ where $C_i \subseteq GF^n(q)$, is the i-th line code, is an (n, m, t, s, M) P-code iff the following conditions hold:

(i) The subspace C_0 is a $(t + s)$-error correcting code.

(ii) For $\mathcal{U}_1$ and $\mathcal{U}_2$, define

$$\mathcal{Z}_{1,2} = \{\boldsymbol{u}_1 - \boldsymbol{u}_2 | \boldsymbol{u}_1 \in \mathcal{U}_1, \boldsymbol{u}_2 \in \mathcal{U}_2\}. \tag{30}$$

Then $\mathcal{Z}_{1,2}$ is a $(2s)$-error correcting code.

(iii) For $i \in \{l(j), l(j) + 1 | j = 2, 3, \ldots, g\}$, the i-th line code C_i is a $(2s)$-error correcting code.

[†]Note that even when m is odd, we can make Type II model include Type I model as a special case by allowing $|\mathcal{L}_g| = 1$.

(iv) For $i \notin \{l(j), l(j)+1 | \, j = 1, 2, \ldots, g\}$, the i-th line code C_i is an s-error correcting code.

(**Proof**) See Appendix A. □

Remark 1: Although Lemma 3 assumes that the first and the second line codes play particular roles for simplicity, we can choose any two consecutive indices $i, i+1$ of lines to impose such roles. Furthermore, the number of such particular line codes is not necessarily two. Even if the number of particular line codes is greater than two, we can discuss similarly. Similar arguments can apply to Lemma 4 stated later, however we assume as in Lemma 3 to simplify the discussion. □

Theorem 4: Let C be an (n, m, t, s, M) P-code with $M = |C|$. We have the following statements:

(i) The size of C is bounded as

$$M \leq A(n, t+s) \times A(n, 2s)^{2g-1} \times A(n, s)^{m-2g}. \tag{31}$$

(ii) For the size $M = A(n, t+s) \times A(n, 2s)^{2g-1} \times A(n, s)^{m-2g}$, there exists an (n, m, t, s, M) P-code.

(**Proof**) It is straightforward to show that Eq. (31) is satisfied from Lemma 3. We will show a construction of an (n, m, t, s, M) P-code whose size achieves (ii). We can take a $(t+s)$-error correcting code with the size $A(n, t+s)$ as C_0 and a $(2s)$-error correcting code with the size $A(n, 2s)$ as $\mathcal{U}_2$ and C_i for $i \in \{l(j), l(j)+1 | \, j = 2, 3, \ldots, g\}$. We set $\mathcal{U}_1 = \{\mathbf{0}\}$. Furthermore, we can take an s-error correcting code with the size $A(n, s)$ as C_i for $i \notin \{l(j), l(j)+1 | \, j = 1, 2, \ldots, g\}$. Apparently, the code C given by $C = C_{1,2} \times \prod_{i=3}^{m} C_i$ satisfies all the condition of Lemma 3, and its size is $M = A(n, t+s) \times A(n, 2s)^{2g-1} \times A(n, s)^{m-2g}$. □

Now we consider LIP-codes C expressed as $C = \prod_{i=1}^{m} C_i$ where the i-th line code $C_i \subseteq GF^n(q)$ is linear.

Lemma 4: A code C is an $(n, m, t, s, \{k_1, k_2, \ldots, k_m\})$ LIP-code iff the following conditions hold:

(i) The subcode $C_0 = C_1 \cap C_2$ is a linear $(t+s)$-error correcting code.

(ii) For C_1 and C_2, the direct sum $C_1 + C_2$ is a linear $(2s)$-error correcting code.

(iii) For $i \in \{l(j), l(j)+1 | \, j = 2, 3, \ldots, g\}$, the i-th line code C_i is a $(2s)$-error correcting code.

(iv) For $i \notin \{l(j), l(j)+1 | \, j = 1, 2, \ldots, g\}$, the i-th line code C_i is an s-error correcting code.

(**Proof**) We can prove the lemma in a similar way to Lemma 3, and hence we omit the proof. □

Theorem 5: For given positive integers n, t, s and $\{k_1, k_2, \ldots, k_m\}$, if there exists an $(n, m, t, s, \{k_1, \ldots, k_m\})$ LIP-code, the dimension tuple $\{k_1, \ldots, k_m\}$ is upper-bounded as

$$\sum_{i=1}^{m} k_i \leq L(n, t+s) + (2g-1)L(n, 2s) + (m-2g)L(n, s). \tag{32}$$

Furthermore, we can obtain the code whose dimension tuple $\{k_1, \ldots, k_m\}$ satisfies

$$\sum_{i=1}^{m} k_i \geq K(n, 2s, t+s) + (2g-1)L(n, 2s) + (m-2g)L(n, s), \tag{33}$$

where $K(n, 2s, t+s)$ denotes the maximum dimension of a linear $(t+s)$-error correcting code contained in a linear $(2s)$-error correcting code with the dimension $L(n, 2s)$.

(**Proof**) Based on the proofs in Theorems 2 and 4, we have Eq. (32). We can also obtain a construction of an $(n, m, t, s, \{k_1, k_2, \ldots, k_m\})$ LIP-code which satisfies Eq. (33) in a similar manner to the proof of Theorem 3. □

6. Discussion about Coding Rate

As mentioned in Sect. 5.1, when m is even and $g = m/2$, the Type II model is identical to the Type I model. Furthermore, we can modify the definition of Type II (t, s)-parallel errors to always include Type I (t, s)-parallel errors as a special case, although we do not modify so to simplify the discussion. From these reasons, in the following, we will discuss coding rate of P-codes for Type II model in main.

Define the **coding rate per a line** of C as

$$R(m) = \frac{\log_q |C|}{nm}. \tag{34}$$

This coding rate indicates the average coding rate per a line. As this rate is greater, we can say that a code becomes more efficient for a fixed m. We will consider the behavior of this coding rate by varying g or m.

[About the Coding Rate with Respect to g]

From Theorem 4, as the number g of disjoint set becomes smaller, the size of a P-code tends to be large for fixed n and m. The size M is the largest if $g = 1$, i.e., the case in which the common error symbols in all lines are identical. In this case, we have

$$R(m) = \frac{1}{nm}\Big\{ \log_q A(n, t+s) + \log_q A(n, 2s) + (m-2) \log_q A(n, s) \Big\}. \tag{35}$$

from Theorem 4.

[About the Coding Rate with Respect to m]

For a given g, as the number of lines, m, becomes greater, the size of a P-code tends to be large. For Type II parallel errors, from Theorem 4, we can have

$$R(m) = \frac{1}{nm}\Big\{ \log_q A(n, t+s) + (2g-1) \log_q A(n, 2s) + (m-2g) \log_q A(n, s) \Big\}. \tag{36}$$

Thus $R(m) = (1 - 2g/m)(\log_q A(n, s))/n + O(m^{-1})$, obtaining

$$\lim_{m\to\infty} R(m) = \lim_{m\to\infty} \left\{ \left(1 - \frac{2g}{m}\right) \frac{\log_q A(n,s)}{n} \right\} = \frac{\log_q A(n,s)}{n}. \quad (37)$$

Equation (37) implies the coding rate per a line tends to close to that of the optimum s-error correcting code for Type II parallel error channel as the number of lines increases. Note that we need to use a $(t+s)$-error correcting code as each line code for correcting any (t,s)-parallel errors when we only adopt ordinary random error correcting codes in respective lines. The coding rate per a line achieves only $(\log_q A(n,t+s))/n$ in this case, which implies the effectiveness of use of parallel error correcting codes.

When m is even and $g = m/2$, Type II model is reduced to Type I model. By a similar argument to Type II case, for Type I case, we obtain $R(m) = (1-1/m)(\log_q A(n,2s))/n + (m^{-1})$ and

$$\lim_{m\to\infty} R(m) = \lim_{m\to\infty} \left\{ \left(1 - \frac{1}{m}\right) \frac{\log_q A(n,2s)}{n} \right\} = \frac{\log_q A(n,2s)}{n}. \quad (38)$$

from Theorem 4. Equation (38) implies the coding rate per a line tends to close to that of the optimum $(2s)$-error correcting code for Type I parallel error channel as the number of lines increases. Note that the coding rate per a line achieves only $(\log_q A(n,t+s))/n$ when we just use ordinary error correcting codes in respective lines, which implies the structure of parallel error correcting codes works well for parallel error channel.

As for LIP-codes, we can discuss similarly based on Theorem 5. Hence we can show the effectiveness of the parallel error correcting codes compared with the case where we just use ordinary error correcting codes in respective lines.

7. Conclusion

In this paper, we generalized the notion of the parallel error channel proposed by Ahlswede et al. by allowing some additional random errors to a conventional parallel error. Then we derived necessary and sufficient conditions for non-independent and linear independent parallel error correcting codes. We showed some construction methods for both non-independent and linear independent codes. Decoding algorithms for these codes are given by combining two kinds of ordinary error correcting codes. Therefore we can find an efficient algorithm for linear parallel error correcting codes. Finally we generalized the results for two lines case to a general $m \geq 3$ lines case. In this case, we considered two types of generalized parallel errors. The obtained result includes the foregoing result for the two lines case as a special case.

As for future works, the probabilistic models of parallel error should be discussed. Conditions of the optimal independent parallel error correcting code for given n, t and s is also to be derived.

Acknowledgments

The authors would like to thank Associate Editor, Prof. M. Mohri, and anonymous reviewers for their valuable comments. This work is supported by Waseda University Grant for Special Research Project No. 2006B-293 and the Telecommunications Advancement Foundation (TAF).

References

[1] R. Ahlswede, B. Balkenhol, and N. Cai, "Parallel error correcting codes," IEEE Trans. Inf. Theory, vol.48, no.4, pp.959–962, April 2002.
[2] S. Chang and E.J. Weldon, Jr., "Coding for T-user multiple-access channels," IEEE Trans. Inf. Theory, vol.IT-25, no.6, pp.684–691, Nov. 1979.
[3] J. Cheng and Y. Watanabe, "A multiuser k-ary code for the noisy multiple-access adder channel," IEEE Trans. Inf. Theory, vol.47, no.6, pp.2603–2607, Sept. 2001.
[4] T. Cover and J. Thomas, Elements of Information Theory, Wiley & Sons, New York, 1991.
[5] R.G. Gallager, Information Theory and Reliable Communication, Wiley & Sons, New York, 1968.
[6] L. Gyorfi and B. Lacmy, "Signature coding and information transfer for the multiple access adder channel," Proc. 2004 IEEE Inf. Theory Workshop, pp.242–246, San Antonio, Texas, Oct. 2004.
[7] T. Kasami and S. Lin, "Bounds on the achievable rates of block coding for a memoryless multiple-access channel," IEEE Trans. Inf. Theory, vol.IT-24, no.2, pp.187–197, March 1978.
[8] R. Liu, P. Spasojevic', and E. Soljanin, "Reliable channel regions for good binary codes transmitted over parallel channels," IEEE Trans. Inf. Theory, vol.52, no.4, pp.1405–1424, April 2006.
[9] F.J. McWilliams and N.J.A. Sloane, The Theory of Error Correcting Codes, North-Holland, Amsterdam, The Netherlands, 1986.
[10] W.W. Peterson and E.J. Weldon, Jr., Error-Correcting Codes, 2nd ed., MIT Press, Cambridge, MA, 1972.
[11] G. Strang, Linear Algebra and Its Applications, Harcourt Brace Jovanovich, San Diego, 1988.
[12] K. Tokiwa, H. Matsuda, and H. Tanaka, "A code construction for M-Choose-T communication over the multiple-access adder channel," IEICE Trans. Fundamentals, vol.E78-A, no.1, pp.94–99, Jan. 1995.
[13] H. Yagi, T. Matsushima, and S. Hirasawa, "A generalization of the parallel error correcting codes," Proc. 2006 IEEE Inf. Theory Workshop, pp.229–233, Chengdu, China, Oct. 2006.

Appendix A: Proof of Lemma 3

First we will prove the if part, assuming that the code C is not an $(n,m,t,s,|C|)$ code. From the assumption that $C = \{(\boldsymbol{c}_1,\boldsymbol{c}_2)\} \times \prod_{i=3}^{m} C_i$, we can consider two distinct codewords $\boldsymbol{c} = (\boldsymbol{c}_1,\boldsymbol{c}_2,\boldsymbol{c}_3,\ldots,\boldsymbol{c}_m) \in C$ and $\boldsymbol{c} = (\boldsymbol{c}'_1,\boldsymbol{c}'_2,\boldsymbol{c}_3,\ldots,\boldsymbol{c}_m) \in C$ where codewords $\boldsymbol{c}_i$ $(i = 3,4,\ldots,m)$ of the last $m-2$ line codes are identical. Supposing two (t,s)-parallel errors $\boldsymbol{e} = (\boldsymbol{e}_1,\ldots,\boldsymbol{e}_m)$ and $\boldsymbol{e}' = (\boldsymbol{e}'_1,\ldots,\boldsymbol{e}'_m)$ have identical error patterns $\boldsymbol{e}_i$ $(i = 3,4,\ldots,m)$ in the last $m-2$ lines, there exist two pairs $(\boldsymbol{e}_1,\boldsymbol{e}_2)$ and $(\boldsymbol{e}'_1,\boldsymbol{e}'_2)$ satisfying

$$\boldsymbol{x} + \boldsymbol{u}_1 + \boldsymbol{e}_1 = \boldsymbol{x}' + \boldsymbol{u}'_1 + \boldsymbol{e}'_1, \quad (A\cdot 1)$$

$$\boldsymbol{x} + \boldsymbol{u}_2 + \boldsymbol{e}_2 = \boldsymbol{x}' + \boldsymbol{u}'_2 + \boldsymbol{e}'_2, \quad (A\cdot 2)$$

where $(\boldsymbol{c}_1, \boldsymbol{c}_2) = (\boldsymbol{x} + \boldsymbol{u}_1, \boldsymbol{x} + \boldsymbol{u}_2)$ and $(\boldsymbol{c}'_1, \boldsymbol{c}'_2) = (\boldsymbol{x}' + \boldsymbol{u}'_1, \boldsymbol{x}' + \boldsymbol{u}'_2)$, since C is not an $(n, m, t, s, |C|)$ code. Subtracting Eq. (A· 2) from Eq. (A· 1), we have

$$\boldsymbol{u}_1 - \boldsymbol{u}_2 - (\boldsymbol{u}'_1 - \boldsymbol{u}'_2) = \boldsymbol{e}'_1 - \boldsymbol{e}'_2 - (\boldsymbol{e}_1 - \boldsymbol{e}_2), \tag{A· 3}$$

and this implies the code $\mathcal{Z}_{1,2}$ is not a $(2s)$-error correcting code, i.e., the condition (ii) does not hold. We can easily prove that the condition (i) does not hold by similar procedure to the proof of Lemma 1.

For some $i^* \in \{l(j), l(j)+1 | \, j = 2, 3, \ldots, g\}$, we consider two codewords $\boldsymbol{c}, \boldsymbol{c}' \in C$ whose i^*-th line codewords $\boldsymbol{c}_{i^*}$ and $\boldsymbol{c}'_{i^*}$ are only distinct, i.e., $\boldsymbol{c}_j = \boldsymbol{c}'_j$ for any $j \neq i^*$. Supposing two (t, s)-parallel errors $\boldsymbol{e}$ and $\boldsymbol{e}'$ whose i^*-th lines' error patterns $\boldsymbol{e}_{i^*}$ and $\boldsymbol{e}'_{i^*}$ are only distinct, we may have

$$\boldsymbol{c}_{i^*} + \boldsymbol{e}_{i^*} = \boldsymbol{c}'_{i^*} + \boldsymbol{e}'_{i^*} \tag{A· 4}$$

since C is not an $(n, m, t, s, |C|)$ code. Subtracting $\boldsymbol{c}_{i^*-1} + \boldsymbol{e}_{i^*-1}$ from both sides of Eq. (A· 4),

$$\boldsymbol{c}_{i^*} - \boldsymbol{c}'_{i^*} = (\boldsymbol{e}'_{i^*} - \boldsymbol{e}_{i^*-1}) - (\boldsymbol{e}_{i^*} - \boldsymbol{e}_{i^*-1}) \tag{A· 5}$$

holds, which implies the i^*-th line code C_{i^*} is not an $(2s)$-error correcting code (note that we here suppose $\boldsymbol{c}_{i^*-1} = \boldsymbol{c}'_{i^*-1}$ and $\boldsymbol{e}_{i^*-1} = \boldsymbol{e}'_{i^*-1}$). Namely, the condition (iii) does not hold.

For some $i^* \notin \{l(j), l(j)+1 | \, j = 1, 2, \ldots, g\}$, by a similar argument, we may have

$$\boldsymbol{c}_{i^*} - \boldsymbol{c}'_{i^*} = \boldsymbol{e}'_{i^*} - \boldsymbol{e}_{i^*}. \tag{A· 6}$$

If $(\boldsymbol{e}_{i^*-2}, \boldsymbol{e}_{i^*-1}) = (\boldsymbol{e}'_{i^*-2}, \boldsymbol{e}'_{i^*-1})$, the common error symbols between $\boldsymbol{e}_{i^*}$ and $\boldsymbol{e}'_{i^*}$ are the same as those between $\boldsymbol{e}_{i^*-2}$ and $\boldsymbol{e}_{i^*-1}$ from the definition of Type II (t, s)-parallel error, i.e., we have

$$\mathcal{T}(\boldsymbol{e}_{i^*}, \boldsymbol{e}'_{i^*}) = \mathcal{T}(\boldsymbol{e}_{i^*-2}, \boldsymbol{e}_{i^*-1}). \tag{A· 7}$$

Thus we can obtain $d_H(\boldsymbol{e}_{i^*}, \boldsymbol{e}'_{i^*}) \leq 2s$, and Eq. (A· 6) indicates the i^*-th line code C_{i^*} is not an s-error correcting code (note that we here suppose $(\boldsymbol{c}_{i^*-2}, \boldsymbol{c}_{i^*-1}) = (\boldsymbol{c}'_{i^*-2}, \boldsymbol{c}'_{i^*-1})$ and $(\boldsymbol{e}_{i^*-2}, \boldsymbol{e}_{i^*-1}) = (\boldsymbol{e}'_{i^*-2}, \boldsymbol{e}'_{i^*-1})$). Namely, the condition (iv) does not hold.

Next we will consider the only-if part, assuming the conditions (i), (ii), (iii) and (iv) do not hold. In this case, we can easily show that there exist codewords $\boldsymbol{c}, \boldsymbol{c}' \in C$ satisfying $\boldsymbol{c} + \boldsymbol{e} = \boldsymbol{c}' + \boldsymbol{e}'$ even if two distinct (t, s)-parallel errors occur. □

Hideki Yagi was born in Yokohama, Japan, on Oct. 14, 1975. He received the B.E. degree, M.E. degree, and Dr.E. degree in Industrial and Management Systems Engineering from Waseda University, Tokyo, Japan, in 2001, 2003 and 2005, respectively. From 2005 to 2007, he was a Research Associate and since 2007, has been an Assistant Professor at Media Network Center, Waseda University, Tokyo, Japan. His research interests are coding theory and information security. He is a member of the Society of Information Theory and its Applications and IEEE.

Toshiyasu Matsushima was born in Tokyo, Japan, on Nov. 26, 1955. He received the B.E. degree, M.E. degree and Dr.E. degree in Industrial and Management Systems Engineering from Waseda University, Tokyo, Japan, in 1978, 1980 and 1991, respectively. From 1980 to 1986, he was with Nippon Electric Corporation, Kanagawa, Japan. From 1986 to 1992, he was a lecturer at Department of Management Information, Yokohama College of Commerce. From 1993, he was an associate professor and from 1996 to 2007 was a professor of School of Science and Engineering, Waseda University, Tokyo, Japan. Since 2007, he has been a professor of School of Fundamental Science and Engineering, Waseda University. His research interests are information theory and its application, statistics and artificial intelligence. He is a member of the Society of Information Theory and Its Applications, the Japan Society for Quality Control, the Japan Industrial Management Association, the Japan Society for Artificial Intelligence and IEEE.

Shigeichi Hirasawa was born in Kobe, Japan, on Oct. 2, 1938. He received the B.S. degree in mathematics and the B.E. degree in electrical communication engineering from Waseda University, Tokyo, Japan, in 1961 and 1963, respectively, and the Dr.E. degree in electrical communication engineering from Osaka University, Osaka, Japan, in 1975. From 1963 to 1981, he was with the Mitsubishi Electric Corporation, Hyogo, Japan. From 1981 to 2007, he was a professor of School of Science and Engineering and since 2007, has been a professor of School of Creative Science and Engineering, Waseda University, Tokyo, Japan. In 1979, he was a Visiting Scholar in the Computer Science Department at the University of California, Los Angels (CSD, UCLA), CA. He was a Visiting Researcher at the Hungarian Academy of Science, Hungary, in 1985, and at the University of Trieste, Italy, in 1986. In 2002, he was also a Visiting Faculty at CSD, UCLA. From 1987 to 1989, he was the Chairman of Technical Group on Information Theory of IEICE. He received the 1993 Achievement Award, and the 1993 Kobayashi-Memorial Achievement Award from IEICE. In 1996, he was the President of the Society of Information Theory and Its Applications (Soc. of ITA). His research interests are information theory and its applications, and information processing systems. He is an IEEE Fellow, and a member of Soc. of ITA, the Operations Research Society of Japan, the Information Processing Society of Japan, the Japan Industrial Management Association, and Informs.