A new Decoding Method of Redundant Residue Polynomial Codes

A New Decoding Method of Redundant Residue Polynomial Codes

Akira SHIOZAKI* and Fujio NISHIDA**

(Received May 29, 1975)

Abstract

This paper presents a new decoding method of redundant residue polynomial codes. The decoding method has an advantage of correcting errors easily by checking the degree of the product of a polynomial corresponding to a received code and the moduli corresponding to error positions. The number of operations needed for this decoding method is about 1/10—1/100 times as large as that needed for the previous methods. Then the number of decoding operations is examined in relation to the construction of the codes in the case of burst-error correction.

1. Introduction

In recent years, many effective codes for multiple-burst-error correction have been proposed. The one is the Reed-Solomon codes (abbreviated to R-S codes) over a finite field, which are also capable of correcting independent errors[1)2)]. The others belong to a class of codes that includes R-S codes as a special case and is constructed using Chinese Remainder Theorem in the residue number system[3)4)]. The efficiency of these codes does not so much get worse even when their code length is long and error-correcting capability is great. Especially, for the correction of both independent errors and burst errors, it is exceedingly good[5)]. Unfortunately, however, their decoding methods are so complicated that the practical use of them seems to be difficult.

This paper proposes a new simple decoding method of multiple-burst-error-correcting codes using the residue number system. These codes can correct phased burst errors or block errors. Each block is expressed as a polynomial over GF(2) or an element over GF(2^m). They can correct t erroneous blocks by the aid of $2t$ redundant blocks. It is shown that they are so efficient that they always meet Varshamov-Gilbert bound[2)] and furthermore approach Gallager bound[2)] asymptotically for burst errors.

2. Residue Number System

Let GF(q) be a finite field with $q=p^m$ elements, GF(q)[x] be a ring of polynomials over GF(q) and any polynomial be the elements of GF(q)[x], where p is a prime

* Graduate Student, Department of Electrical Engineering, Collage of Engineering.
** Department of Electrical Engineering, Collage of Engineering.

number.

(Definition 1) Two polynomials $m_1(x)$ and $m_2(x)$ are said to be relatively prime if and only if $g(x)|m_1(x)$*, $g(x)|m_2(x)$, and $g(x)$ is a constant polynomial.

(Definition 2) Two polynomials $a(x)$ and $b(x)$ are said to be congruent for the modulus $m(x)$ or simply congruent modulo $m(x)$, denoted $a(x) \equiv b(x) \mod m(x)$, if and only if $m(x)|(a(x)-b(x))$.

(Chinese Remainder Theorem) Let $m_1(x)$, $m_2(x)$, ..., and $m_r(x)$ be relatively prime in pairs and $M(x)$ denote their product $\prod_{i=1}^{r} m_i(x)$. If $a_1(x)$, $a_2(x)$, ..., and $a_r(x)$ are any given polynomials, then there exists one and only one member $f(x)$ of GF(q)[x] such that

$$\deg[f(x)] < \deg[M(x)]^{**} \tag{1}$$

and

$$f(x) \equiv a_i(x) \mod m_i(x) \quad (i=1, 2, \ldots, r). \tag{2}$$

Let $t_i(x)$ be a polynomial satisfying

$$\frac{M(x)}{m_i(x)} t_i(x) \equiv 1 \mod m_i(x) \quad (i=1, 2, \ldots, r) \tag{3}$$

The existence of such polynomials is assured by the assumed relative primeness property of the $m_i(x)$. Then $f(x)$ is given by the following equation:

$$f(x) \equiv \frac{M(x)}{m_1(x)} t_1(x)a_1(x) + \frac{M(x)}{m_2(x)} t_2(x)a_2(x) + \ldots + \frac{M(x)}{m_r(x)} t_r(x)a_r(x) \mod M(x). \tag{4}$$

The algorithm of constructing $t_i(x)$ is stated in Ref. (3) in detail.

3. Code Construction and Decoding Method

The decoding method mentioned here is similar to the error correcting method in the residue number system using integers[6)7)]. The following discussions in this paper are confined to q=2.

Let $m_1(x)$, $m_2(x)$, ..., and $m_n(x)$ be n moduli which are relatively prime and $M(x)$ denote $\prod_{i=1}^{n} m_i(x)$. Furthermore, assume that the degree of each $m_i(x)$ is d and kd information symbols $u=(u_0, u_1, \ldots, u_{kd-1})$ are represented by a polynomial:

$$F(x) = u_0 + u_1x + \ldots + u_{kd-1}x^{kd-1}. \tag{5}$$

Then in place of the original block u, the coefficients of $a_i(x)$ are sent in order as follows:

---

* $g(x)|m_1(x)$ denotes $g(x)$ divides $m_1(x)$.

** $\deg[f(x)]$ denotes the degree of $f(x)$.

$$v=(a_1(x), a_2(x), \ldots, a_n(x)), \tag{6}$$

where $$a_i(x)\equiv F(x) \quad mod\ m_i(x) \quad (i=1, 2, \ldots, n) \tag{7}$$

and deg $[a_i(x)]<d$. The vector v is called the residue representation of $F(x)$, and the vector corresponding to the polynomial $a_i(x)$ is named the i-th block.

Next we show a decoding method. Assume that the l_1-th, the l_2-th, ..., and the l_t-th blocks are erroneous and the received code v' is

$$v'=(a_1'(x), a_2'(x), \ldots, a_n'(x)). \tag{8}$$

Let an error vector be

$$e=(0, \ldots, e_{l_1}(x), \ldots, 0, \ldots, e_{l_t}(x), \ldots, 0). \tag{9}$$

Then the following relations hold:

$$\left.\begin{array}{l} a_{l_i}'(x)\equiv a_{l_i}(x)+e_{l_i}(x) \quad mod\ m_{l_i}(x) \quad (i=1, 2, \ldots, t), \\ a_j'(x)=a_j(x) \quad (j\neq l_i). \end{array}\right\} \tag{10}$$

Let $F'(x)$ and $E(x)$ (deg $[F'(x)]<nd$, deg $[E(x)]<nd$) be polynomials whose residue representations by modulo $m_i(x)$ are Eqs. (8) and (9) respectively. Then, from Eq. (4), $F'(x)$ and $E(x)$ are represented as follows:

$$F'(x)\equiv\sum_{i=1}^{n}\frac{M(x)}{m_i(x)}\,t_i(x)a_i'(x) \quad mod\ M(x), \tag{11}$$

$$E(x)\equiv\sum_{i=1}^{t}\frac{M(x)}{m_{l_i}(x)}\,t_{l_i}(x)e_{l_i}(x) \quad mod\ M(x) \tag{12}$$

and

$$F'(x)\equiv F(x)+E(x) \quad mod\ M(x). \tag{13}$$

Let us define

$$B(x)\equiv\sum_{i=1}^{t}\frac{D(x)}{m_{l_i}(x)}\,t_{l_i}(x)e_{l_i}(x) \quad mod\ D(x), \tag{14}$$

where $D(x)=\prod_{i=1}^{t} m_{l_i}(x)$, then from Eq. (12), $E(x)$ becomes

$$E(x)=\frac{M(x)}{D(x)}\sum_{i=1}^{t}\frac{D(x)}{m_{l_i}(x)}\,t_{l_i}(x)e_{l_i}(x)=\frac{M(x)}{D(x)}B(x), \tag{15}$$

where $$\deg\,[B(x)]<\deg\,[D(x)]. \tag{16}$$

Thus, both Eqs. (15) and (16) lead to

$$(n-t)d\leqq \deg\,[E(x)] < nd. \tag{17}$$

Eq. (13) yields

$$F'(x)=F(x)+E(x) \tag{18}$$

and

$$(n-t)d \leqq \deg [F'(x)] < nd. \tag{19}$$

On the other hand, Eq. (5) assures

$$0 \leqq \deg [F(x)] < kd. \tag{20}$$

Consequently, if $kd \leqq (n-t)d$, that is

$$n-k \geqq t, \tag{21}$$

the next theorem is obtained from Eqs. (19) and (20).

(Theorem 1) If $n-k \geqq t$, a code vector obtained from Eq. (6) can detect errors less than or equal to t blocks as follows:

(i) If $0 \leqq \deg [F'(x)] < kd$, no error occurs.

(ii) If $kd \leqq \deg [F'(x)] < nd$, errors less than or equal to t blocks occur.

Besides, the following several propositions hold:

(Proposition 1)

$$d \leqq \deg [D(x)F'(x) \quad mod\ M(x)] < (k+t)d. \tag{22}$$

(Proof) The equality

$$F'(x) = F(x) + \frac{M(x)}{D(x)} B(x)$$

yields

$$D(x)F'(x) = D(x)F(x) + M(x)B(x) \equiv D(x)F(x) \quad mod\ M(x). \tag{23}$$

Then, since $0 \leqq \deg [F(x)] < kd$ and $d \leqq \deg [D(x)] \leqq td$, Eq. (22) is derived. QED

(Proposition 2)

$$d \leqq \deg [D^*(x)F'(x)\ mod\ M(x)] < (k+t)d, \tag{24}$$

where $D^*(x) = D(x)P(x)$ and $\deg [D^*(x)] = td$.

(Proof) Since $0 \leqq \deg [F(x)] < kd$, $\deg [D^*(x)] = td$ and

$$D^*(x)F'(x) = D^*(x)F(x) + P(x)M(x)B(x) \equiv D^*(x)F(x) \quad mod\ M(x),$$

Eq. (24) is derived. QED

(Proposition 3)

$$(n-t)d \leqq \deg [D'(x)F'(x) \quad mod\ M(x)] < nd \tag{25}$$

where $D'(x)$ denotes the product of moduli such that $D(x) \nmid D'(x)$†.

(Proof) The equality

$$D'(x)F'(x) = D'(x)F(x) + D'(x) \frac{M(x)}{D(x)} B(x)$$

yields

† $D(x) \nmid D'(x)$ denotes $D(x)$ does not divide $D'(x)$.

$$D'(x)F'(x) \quad mod\ M(x)$$

$$= D'(x)F(x) + D'(x)\frac{M(x)}{D(x)}B(x) - M(x)Q(x)$$

$$= D'(x)F(x) + \frac{M(x)}{D(x)}\{D'(x)B(x) - D(x)Q(x)\}, \qquad (26)$$

where $Q(x)$ is a quotient of $D'(x)M(x)B(x)/D(x)$ when divided by $M(x)$. Eq. (26) and the following three relations:

$$D'(x)B(x) - D(x)Q(x) \neq 0,$$

$$0 \leqq \deg\,[D'(x)B(x) - D(x)Q(x)] \leqq \deg\,[D(x)] - 1$$

and

$$d \leqq \deg[D'(x)F(x)] < (k+t)d$$

derive Eq. (25). QED

Thus, if $(k+t)d \leqq (n-t)d$, that is

$$n-k \geqq 2t, \qquad (27)$$

the next theorem is obtained from Eqs. (24) and (25).

(Theorem 2) If $n-k \geqq 2t$, a code vector constructed from Eq. (6) can correct errors less than or equal to t blocks as follows:

(i) If $0 \leqq \deg\,[F'(x)] < kd$, no error occurs.

(ii) If $kd < (n-t)d \leqq \deg\,[F'(x)] < nd$, construct the product of t moduli, $D^*(x) = \prod_{i=1}^{t} m_{l_i}(x)$, such that it satisfies the inequality

$$d \leqq \deg\,[D^*(x)F'(x)\ mod\ M(x)] < (k+t)d. \qquad (28)$$

Then, using the received polynomial $F'(x)$, the information part of the code is correctly decoded as

$$F(x) = \frac{D^*(x)F'(x) \quad mod\ M(x)}{D^*(x)}. \qquad (29)$$

(iii) If $D^*(x)$ that satisfies Eq. (28) does not exist, there must be uncorrectable errors.

4. Extension to Galois Feild GF(2^m)

In the preceding sections, encoding and decoding are described in the ring of polynomials over GF(2). In this section it is discussed that the same methods can be applied to the ring of polynomials over GF(2^m).

First encoding is mentioned. Let α be a primitive element in an extension field GF(2^m) and let 2^m polynomials which are relatively prime be

$$\begin{aligned} &m_1(x)=x,\\ &m_2(x)=x-1,\\ &m_3(x)=x-\alpha,\\ &\cdots\cdots\cdots\cdots\\ &m_n(x)=x-\alpha^{2^m-2}\\ &\qquad (n=2^m). \end{aligned}$$

Denote k information symbols $(u_0, u_1, \ldots, u_{k-1})$, $u_i \in \mathrm{GF}(2^m)$, by a polynomial representation as follows:

$$F(x)=u_0+u_1x+\ldots+u_{k-1}x^{k-1}.$$

Then, the equations

$$\begin{aligned} &F(0)\equiv F(x) \quad mod\ m_1(x),\\ &F(1)\equiv F(x) \quad mod\ m_2(x),\\ &F(\alpha)\equiv F(x) \quad mod\ m_3(x),\\ &\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\\ &F(\alpha^{2^m-2})\equiv F(x) \quad mod\ m_n(x) \end{aligned}$$

lead to a representation of a code vector v:

$$v=(F(0), F(1), F(\alpha), \ldots, F(\alpha^{2^m-2})).$$

These are Reed-Solomon codes.

Decoding can be performed by letting the degree of $m_i(x)$ be one for all i in the decoding mentioned in Section 3. Now, the residue of $M(x)/m_i(x)$ modulo $m_i(x)$ becomes

$$\begin{aligned} \left(\frac{M(x)}{m_i(x)} \quad mod\ m_i(x)\right) &= \alpha^0\cdot\alpha^1\cdot\alpha^2 \ldots \alpha^{2^m-2}\\ &= \alpha^{\frac{(2^m-2)(2^m-1)}{2}}\\ &= \alpha^{(2^{m-1}-1)(2^m-1)}\\ &= 1 \end{aligned}$$

$$(i=1, 2, \ldots, n), \tag{30}$$

where $M(x)=\prod_{i=1}^{n} m_i(x)$. Then, the equation

$$\frac{M(x)}{m_i(x)}\, t_i(x)\equiv 1 \quad mod\ M_i(x) \quad (i=1, 2, \ldots, n) \tag{31}$$

states

$$t_i(x)=1 \quad (i=1, 2, \ldots, n) \tag{32}$$

and, since deg $[M(x)/m_i(x)] <$ deg $[M(x)]$,

$$F(x)=\sum_{i=1}^{n} a_i \frac{M(x)}{m_i(x)}\ , \tag{33}$$

where $a_i \equiv F(x) \ \ mod\ m_i(x) \ \ (i=1, 2, \ldots, n)$.
Therefore the operation modulo $M(x)$ can be omitted when $F(x)$ is recaptured.

5. Discussion

5.1. A Class of Codes

The number of moduli $m_i(x)$ which decide each component $a_i(x)$ of a code vector cannot be arbitrarily chosen. The maximum number of moduli $m_i(x)$ which can be chosen is restricted by the degree d of $m_i(x)$ since each modulus is relatively prime in pairs. Table 1 illustrates the maximum number n of moduli, which assigns the maximum code length, versus the block length d in the polynomial ring over GF(2).

The maximum code length n is equal to 2^m when the codes are constructed over GF(2^m). Table 2 shows the maximum code length n versus the block length m in the polynomial ring over GF(2^m).

Table 1. Maximum code length versus block length (over GF(2))

block length d (bit)	maximum code length n (block)
3	4
4	6
5	9
6	12
7	23
8	37

Table 2. Maximum code length versus block length (over GF(2^m))

block length m (bit)	maximum code length n (block)
3	8
4	16
5	32
6	64
7	128

5.2. Burst-Error-Correcting Capability and Number of Operations Needed for Decoding

As mentioned previously, the codes which correct block errors are also effective for burst-error correction, especially multiple-burst. This section discusses the error-correcting capability and the number of operations needed for decoding of several codes which are constructed for burst-error correction.

5.2.1. *Operations required for the decoding process*

The operations needed for decoding consist of the following three steps.

(i) The operations to recapture $F'(x)$ from the received code vector $(a'_1(x), a'_2(x), \ldots, a'_n(x))$ using the predetermined value of $M(x)t_i(x)/m_i(x)$ *mod* $M(x)$ as follows:

$$F'(x) \equiv \sum_{i=1}^{n} \frac{M(x)}{m_i(x)} t_i(x) a_i'(x) \qquad mod\ M(x).$$

(ii) The operation of $D^*(x)F'(x)$ *mod* $M(x)$.

(iii) The operation of $F(x) = \dfrac{D^*(x)F'(x)\ mod\ M(x)}{D^*(x)}$.

5.2.2. *The case of using multiple-burst-error-correcting codes*

The codes which correct errors less than or equal to t blocks can also correct s-fold multiple burst errors with a total length b such that

$$b = d(t-s)+s \tag{34}$$

where d is a block length. Especially, for a single burst error, the correctable burst length is

$$b = d(t-1)+1. \tag{35}$$

In the case of correcting multiple burst errors, the number of operations needed for decoding is equal to that required to correct errors less than or equal to t blocks. The number of operations needed for decoding is as follows. In step (i), n times of multiplication of $M(x)t_i(x)a_i'(x)/m_i(x)$ and one time of calculation of the residue modulo $M(x)$ are necessary. In step (ii), the number of selecting $D^*(x)$ is $\binom{n}{t}$ and for each $D^*(x)$, t times of multiplication and one time of calculation of the residue modulo $M(x)$ are required. So at most $\binom{n}{t}\cdot(t+1)$ times of operations are necessary. In step (iii), one time of division by $D^*(x)$ is required. Therefore, the total number of operations is at most

$$n+2+\binom{n}{t}\cdot(t+1). \tag{36}$$

In the case of correcting a single burst error, $(n+1)$ times of operations are needed for step (i). In step (ii), as the number of selecting $D^*(x)$ is n regardless of t, the number of operations is at most $n(t+1)$. And in step (iii) that is one. Therefore, the total number of operations is at most

$$n(t+2)+2. \tag{37}$$

5.2.3. *The case of interleaving single-error-correcting codes*

Now we briefly explain an interleaving method. The method mentioned here is one of constructing burst-error-correcting codes with large code length by concatenating some single-error-correcting codes with small code length.

Let s single-error-correcting codes be

$$(a_{11},\ a_{12},\ \ldots,\ a_{1n})$$
$$(a_{21},\ a_{22},\ \ldots,\ a_{2n})$$
$$\cdots\cdots\cdots\cdots\cdots\cdots$$
$$(a_{s1},\ a_{s2},\ \ldots,\ a_{sn}).$$

By concatenating these codes, a transmitting code

$$(a_{11},\ a_{21},\ \ldots,\ a_{s1},\ a_{12},\ a_{22},\ \ldots,\ a_{s2},\ \ldots,\ a_{1n},\ a_{2n},\ \ldots,\ a_{sn})$$

is constructed. This code can correct a single burst error whose length is

$$b=d(s-1)+1, \tag{38}$$

where d is a block length. The number of operations needed for decoding is

$$s(3n+2) \tag{39}$$

because it is s times of the number of operations for a single-error-correcting code.

5.2.4. *Comparison between two cases for single-burst-error correction*

In this section, some comparisons are made between two kinds of single-burst-error-correcting codes described in the preceding section, where one code, denoted [a], uses a multiple-error-correcting code, while the other, denoted [b], interleaves single-error-correcting codes. In the comparison, the block length, the code length and the error-correcting capability of the two codes are constrained to be the same. Table 3 illustrates the comparison between the number of operations required to decode the code [a] and that to decode the code [b].

Table 3. Comparison of the numbers of operations needed for decoding (each block has 8 bits)

code length n(bit)	information bits k	correctable burst-length b (bit)	the number of operations	
			[a]	[b]
96	48	17	62	42
128	64	25	98	56
160	80	33	142	70
192	96	41	194	84
256	128	57	322	112

As shown in Table 3, single-burst-error correction using the method interleaving single-error-correcting codes needs smaller number of operations than that using a multiple-error-correcting code. Moreover, as shown in Table 1, the use of a multiple-error-correcting code cannot provide the construction of a long code, while the use of the interleaving methods are capable of providing the construction of an arbitrarily long code by increasing the interleaving number s. Therefore the method of interleaving single-error-correcting codes is much preferable for a single-burst-error-correction.

5.3. Two Kinds of Constructions of Codes with the Same Length

When the codes with the constant code length n and the constant error-correcting capability t are constructed, the next two cases of selecting the moduli are considered.

[1] The case of using a few moduli with large degree.

[2] The case of using many moduli with small degree.

The number of operations needed for encoding and decoding in the case [1] is smaller than that in the case [2] because the number of moduli is small. On the other hand, the redundancy in the case [2] is less than that in the case [1]. For the burst-error correction, the correctable burst-error length in the case [1] is longer than that in the case [2], but the ratio of the correctable burst-length to the redundancy in the case [1] is lower than that in the case [2]. For example, consider the next two codes: [1] a code which consists of 9 blocks (each block has 8 bits) and corrects two block's errors and [2] a code which consists of 12 blocks (each block has 6 bits) and corrects two block's errors. The comparison of some characteristics of these codes is shown in Table 4.

Table 4. Comparison between code [1] and code [2]

	code [1]	code [2]
code length n (block)	9	12
block length d (bit)	8	6
error-correcting capability t (block)	2	2
rate k/n	0.56	0.67
correctable burst-length b (bit)	9	7
$b/(n-k)$	0.28	0.29
maximum number of operations needed for decoding	38	50

5.4. Comparison of the Numbers of Operations Needed for Decoding

This section describes the outline of the decoding method in Ref. (3) and shows the comparison between the numbers of operations needed for decoding by the method in Ref. (3) and that in this paper.

Let the received code vector be $(a_1'(x), a_2'(x), \ldots, a_n'(x))$ and the number of blocks of information be k. The decoding procedure is as follows:

(i) Select arbitrary k residue digits $a_{l_i}'(x)$ $(i=1,2,\ldots, k)$ from the received code vector and recapture the polynomial $f(x)$ corresponding to the residue representation with k residue digits.

(ii) Perform similar calculations for all combinations of k residue digits.

(iii) For $\binom{n}{k}$ recaptured polynomials, if all recaptured polynomials agree with each other, decide that no error occurs.

(iv) If not all recaptured polynomials agree with each other, let the polynomial which has a majority be the information polymomial.

In the decoding method in Ref. (3), the total number of operations needed for decoding is

$$(k+1)\cdot\binom{n}{k} \tag{40}$$

since the number of operations of recapturing each $f(x)$ is $(k+1)$. The number of operations is the same for both the multiple-burst-error correction and the single-burst-error correction because the method in Ref. (3) is based on the majority decoding.

Table 5 shows the comparison between the number of operations in the decoding method [a] proposed in this paper and that in the method [b] in Ref. (3).

Table 5. Comparison of the numbers of operations needed for decoding

code length n (block)	information length k (block)	correcting capability t (block)	the number of operations [a]		the number of operations [b]
			multiple burst error	single burst error	
10	8	1	32	32	405
	6	2	147	42	1470
15	11	2	332	62	16380
	9	3	1837	77	50050
	7	4	6842	92	51480
20	14	3	4582	102	581400
	12	4	24247	122	1637610
	10	5	93046	142	2032316

As seen from Table 5, the number of operations needed for decoding by the method presented in this paper is about 1/10～1/100 times for multiple-burst-error correction and about 1/1000～1/10000 times for single-burst-error correction as large as one needed for the method in Ref. (3).

6. Conclusion

A new decoding method of the redundant residue polynomial codes has been proposed. This decoding method has the advantage of correcting errors easily by checking the degree of the product of the polynomial corresponding to a received code and the moduli corresponding to the error positions. The number of operations required for decoding by this method is about 1/10～1/100 times as large as that needed for the previous methods. For single-burst-error correction in this method, the use of interleaving single-error-correcting codes is much preferable to that of a multiple-error-correcting code from a point of view of the number of operations needed for decoding as well as the number of classes of codes.

A code in the ring of polynomials over GF(2) cannot have so long code-length. But this disadvantage is removed by constructing a code over GF(2^m). A code over GF(2^m) can be decoded in a similar way to a code over GF(2). When codes are constructed over GF(2^m), the matter of facility of implementation is an open question.

References

1) I.S. Reed and G. Solomon, J. SIAM **8**, 300 (1960).
2) W.W. Peterson and E.J. Weldon, Error-correcting codes, Second edition, Cambridge, Mass., M.I.T. Press (1972).
3) J.J. Stone, J. SIAM **11**, 74 (1963).
4) D. Mandelbaum, IEEE Trans., IT–**14**, 518 (1968).
5) H. Tanaka, S. Okasaka, Y. Tezuka and Y. Kasahara, IECE of Japan **52–A**, 408 (1969).
6) A. Shiozaki, F. Nishida and H. Tanaka, Lecture in Annual Meeting of IECE of Japan (July 1974).
7) A. Shiozaki, F. Nishida and H. Tanaka, Report of Communication System Group, IECE of Japan, **CS74–74** (Sept. 1974).