90° - Rotationally Invariant Multilevel Convolutionally Encoded QAM

Werner Henkel, Michael Koch(*)
Research Centre of Deutsche Bundespost Telekom
P.O. Box 10 00 03, D-6100 Darmstadt

Abstract. A multilevel convolutional code construction is proposed that leads to low complexity rotationally invariant schemes with coding gains of around 6 dB. Conditions for 90° - rotational invariance are derived, and a special structure for differential en/decoding is given. These considerations together with the introduction of a relation between information sequences, when determining the code generators, are the basis for the approach.

1. INTRODUCTION

In order to decrease the length of error bursts caused by carrier phase instabilities (random walks), several proposals have been made to define rotationally invariant coded modulation. Some were based on trellis codes (multidimensional or nonlinear), others on multilevel coding schemes. Such multilevel constructions with rotational-invariance properties were only studied with component codes chosen to be block codes. ([1, 2, 3]). Especially, 90° - invariant block-coded QAM was treated in [4]. Conditions for rotational invariance were derived, and examples with coding gains of up to 6 dB were given. These codes offer a high code rate, enabling to apply e.g. RS codes as additional outer codes. This leads to a further increase in coding gain. All the examples are based on Reed-Muller codes. Although there exist trellis-decoders for maximum likelihood decoding, the question arose, whether the decoding complexity could be reduced by choosing the component codes to be binary convolutional codes. Such schemes without any rotational-invariance properties have already been studied in [5].

In this paper we will derive conditions for the convolutional component codes and describe a construction principle that ensures the QAM to be 90° - invariant.

First of all, conditions are stated that guarantee the rotational invariance of the codes, not yet of the information part. This is done in a following section, introducing a special differential en- and decoding.

(*) Mr. Koch is now with Siemens AG, Dept. ÖV, D - 8000 München

2. CONDITIONS FOR THE BINARY COMPONENT CODES

Multilevel codes protect the labelling of the set partitions of a modulation alphabet against errors. This means that m binary (convolutional) codes are needed to select the points from the $M = 2^m$ -QAM. Let the code rates be

$$R^{(j)} = \frac{k^{(j)}}{n^{(j)}}, \; j = 1, \ldots, m;$$

$k^{(j)}$ denoting the number of input bits and $n^{(j)}$ labelling the number of output bits per frame of a binary convolutional coder. In the case of block codes $n^{(j)}$ is the length of the codewords and $k^{(j)}$ is the number of information bits in a codeword or the dimension of the code. For convenience, we assume all $n^{(j)}$ to be equal ($= n$).

The columns of a matrix with m infinitely long rows, which are the code sequences of the binary convolutional codes, select the points of a 2^m -QAM. A component of the first row (of the first component code) $(a_i^{(1)})$ determines a 2^{m-1} -*QAM* subset, a component of the second $(a_i^{(2)})$ determines a 2^{m-2} -*QAM* subset of the 2^{m-1} -*QAM* set, and so on. Figs. 1 and 2 illustrate the procedure in the case of 16-QAM.

Here, slanted '*QAM*' is used for subsets that are not necessarily rectangular or symmetrically bounded regions of scaled and shifted Z^2, although the considered partitioned QAM has the usual rectangular or cross configuration beginning with 16-QAM (16: rect., 32: cross, 64: rect.,...). An 8-*QAM* set, which can easily be partitio-

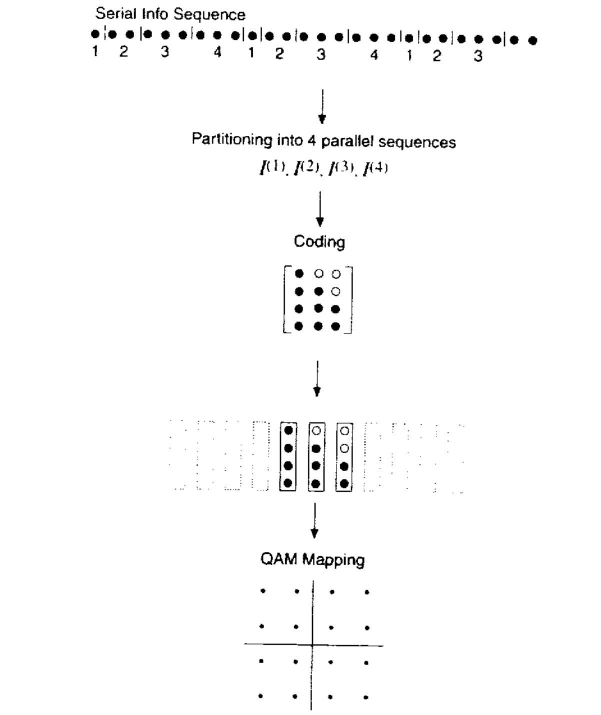

Fig. 1 - Multilevel encoding procedure for convolutionally encoded 16 -QAM (1-st stage: code rate $R^{(1)} = 1/3$, 2-nd stage: code rate $R^{(2)} = 2/3$, 3-rd and 4-th stage uncoded).

ned into two 4 -*QAM* subsets is given in Fig. 2.

The minimum squared Euclidean distance of two sequences of such a multilevel code is known to be

$$d_{E_{\min}} = \min_{j}\left\{d_H^{(j)} \cdot d_E^{(j)}\right\} \qquad (1)$$

where $d_H^{(j)} = d_f^{(j)}$

is the free (Hamming) distance of the j-th component code and $d_E^{(j)}$ is the minimum squared Euclidean distance between the corresponding $2^{(m-j)}$ -*QAM* subsets. From [4] we know that it is mandatory to choose the labelling

$$a_i^{(j)}, j = 3, \ldots, m$$

to be 90° - invariant itself (Fig. 2).

It has already been stated in [4] (for the block-coded case) that a 90° - phase shift results in an inversion of all components of the first codeword, whilst the second codeword is altered by a componentwise addition with the (original) first codeword. Components of the following codes remain unchanged, which is due to the special rotationally invariant labelling.

The conditions that follow are more or less the same as in the block-coded case:

1. The all-ones sequence must be a valid code sequence of code (1). $(\ldots,1,1,\ldots,1,\ldots) \in \mathcal{A}^{(1)}$
2. All valid code sequences of code (1) must be valid code sequences of code (2), too. $\mathcal{A}^{(1)} \subset \mathcal{A}^{(2)}$
3. No conditions for $\mathcal{A}^{(j)}$, $j \geq 3$, where $\mathcal{A}^{(j)}$ denotes the j-th component binary convolutional code (of infinite length). The codes are assumed to be linear.

These conditions ensure the 90° - shift invariance of the multilevel code, which means that valid code sequences turn into (other) valid code sequences. This does not yet guarantee that the information part is maintained, too.

To obtain 90° - invariance with respect to the information, a special differential en/decoding scheme is necessary, which is described below.

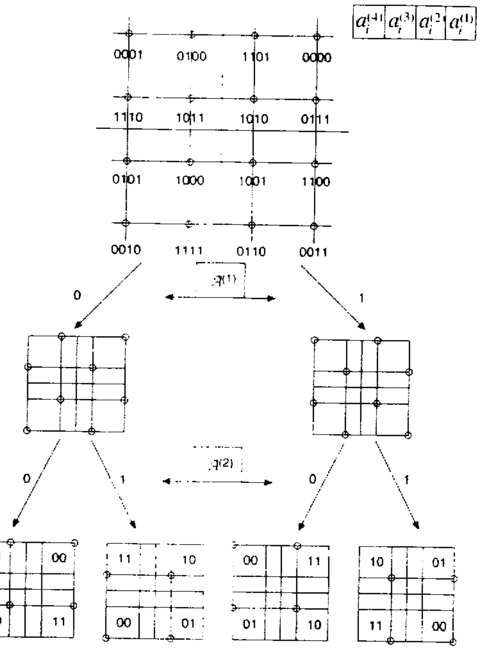

Fig. 2 - Set partitions of 16 - QAM.

3. POSSIBILITIES FOR DIFFERENTIAL EN - AND DECODING

Figs. 3 and 4 show possible configurations for differential en- and decoding, respectively. The structure of the decoder is much simpler than the encoder. Therefore, we first focus on this part.

The differential decoder is located *after* the multistage convolutional decoder. This is necessary, because otherwise a 3 dB loss will be introduced by the differential operation, decreasing achievable coding gain. Of course, single error events will double, but this will reduce the coding gain only slightly at interesting bit-error probabilities. Note that bit-error curves over E_b/N_0 become steeper with decreasing error probability. A small factor (slightly more than two, because the set-partitioning does not lead to a Gray coding of the first two bits) in the error probability will not influence the coding gain too much in areas, where the error curve is rather steep.

The differential encoder might seem somewhat mysterious. It is located between the encoding stages one and two. As will be seen, it is essential for this approach to have a systematic second-level code. This can be easily fulfilled, because for each binary linear convolutio-

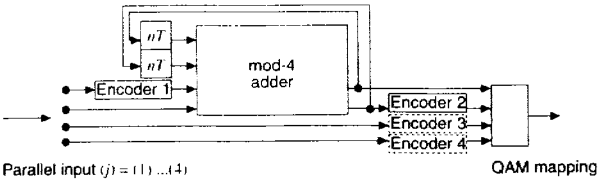

Fig. 3 - Differential encoding for 90° - phase invariant multilevel encoded 16 - QAM.

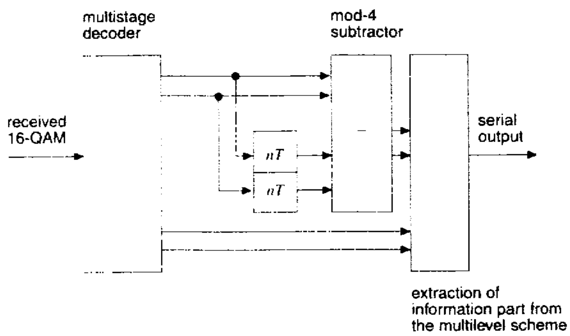

Fig. 4 - Differential decoding for 90° - phase invariant multilevel encoded 16 - QAM.

nal code an equivalent systematic encoder can be derived by means of trivial operations on the rows of the Forney generator matrix ([6, 7, 8]).

Let us now first consider what happens to the components of the first code. Concerning code (1), we may illustrate the differential encoding as a modulo-2 addition of an infinite number of shifted (by n) versions of one sequence (Fig. 5).

Assuming code (1) to be linear, differential encoding leads to another valid code sequence. This allows the multistage decoder to precede the differential decoder at the receiver.

The stage-2 encoder and decoder are next to the channel and thus, no additional conditions (beyond the ones for rotational invariance of the code) are to be fulfilled by code (2). (Compare the differential encoder in [4]). Of course, it is not guaranteed that the differential decoding after decoding of code (2) yields another valid code sequence of code (2). Nevertheless, if we choose code (2) to be systematic, the differential decoding will at least deliver a valid information sequence. Thus, as usual, we simply have to discard the parity components of code (2) after differential decoding.

Although, only code (2) has to be systematic, for convenience, we propose to choose the corresponding systematic (recursive) encoders for both codes (1) and (2).

Components of the codes (3),(4),... are in no way affected by the differential en/decoding. This corresponds to the fact that only code-(1)/(2) components are changed by 90° - phase shifts.

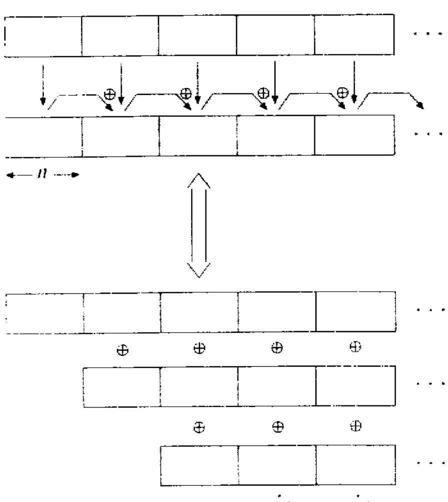

Fig. 5 - An illustration of blockwise (blocklength n) differential encoding modulo 2.

4. MEASURES TO ENSURE THAT THE CONDITIONS FOR ROTATIONAL INVARIANCE ARE MET

As outlined in section 2, the necessary and sufficient condition for phase invariance with respect to multiples of $\pi/2$ for code (1) is the property that the infinite all-ones sequence is a valid code sequence. This is ensured, if there is a loop in the state diagram with an all-ones output. From an own computer search, we know that there are no such codes with even *and* odd weighted generators(1) for $k = 1$. Codes with only even weighted generators are catastrophic, because there is a distance-zero loop at the all-ones state. Thus, code (1) with $k^{(1)} = 1$ has to consist only of odd weighted generators. Considering another coding scheme, Saito [9] came to the same conclusion.

For $k^{(1)} \geq 1$, the all-ones code sequence can be obtained, if there is the possibility of creating odd weighted generators by combining some rows of the Forney matrix of code (1). This, for example, is fulfilled, if one row consists only of odd weighted generator polynomials or if each entry of the generator matrix is odd weighted.

To ensure rotational invariance for code (2), as a necessary and sufficient condition one has to ensure, that every valid code sequence $A^{(1)} = I^{(1)} \cdot G^{(1)}$ of code (1) is also belonging to the set of code sequences $A^{(2)} = I^{(2)} \cdot G^{(2)}$ of code (2). ($G^{(j)}$ denotes the Forney generator matrix of Code (j).

$$\forall_{I^{(1)}} \exists_{I^{(2)}} : I^{(2)} \cdot G^{(2)} = I^{(1)} \cdot G^{(1)} \qquad (2)$$

As this equation has to be fulfilled for arbitrary $I^{(1)}$, $I^{(2)}$ is a function of $I^{(1)}$

$$I^{(2)} = I^{(1)} G^{(1)} \left(G^{(2)} \right)^{-1} \qquad (3)$$

A possible approach for the construction of code (2) is to define the components of

$$I^{(2)} = \left(I_1^{(2)}, I_2^{(2)}, \ldots I_{k^{(2)}}^{(2)} \right)$$

(1) The term generator characterizes one of the n modulo-2 adders. Thus, one column of the Forney matrix is a generator.

as shifted versions of $I^{(1)}$ (assuming $k^{(1)} = 1$):

$$I_h^{(2)} = I^{(1)} \cdot D^{j_h} \cdot$$

$$\left(k^{(1)} = 1, h = 1, \dots k^{(2)}, j_h \in \left\{0, \dots, L^{(1)} - 1\right\}\right) \quad (4)$$

$k^{(j)}$ is the number of information bits of code (j) (per frame). $L^{(j)}$ is the constraint length(²) (memory plus one) of code (j). D is a time delay factor (z^{-1} of the Z - transform).

There has to be at least one

$$I_h^{(2)} = I^{(1)}, \text{i.e., } j_h = 0,$$

in order to express the low-order term $D^0 = 1$, appearing in $G^{(1)}$, by means of $G^{(2)}$.

Defining

$$I^{(2)} \text{ by } I_1^{(2)} = I^{(1)},\ I_h^{(2)} = 0 \left(h = 2, \dots, k^{(2)}\right)$$

leads to a construction where the Forney matrix of code (1) is one row of the code (2) matrix. The other rows can be chosen freely according to a maximization of free distance.

This simple approach is disadvantageous as it requires $L^{(2)} \geq L^{(1)}$. For a better relation between coding gain and decoding complexity one should choose $L^{(2)} \leq L^{(1)}$, as the decoding complexity is proportional to $2^{k \cdot L}$. Note that the number of possible paths in the Trellis of the Viterbi algorithm equals the product of the number of the states ($2^{k(L-1)}$) and the number of the paths starting from one state (2^k). Furthermore, a higher constraint length for code (2) seems not to be necessary, because in order to ensure a nearly constant distance product $d_H^{(j)} \cdot d_E^{(j)}$, a decreasing Hamming or free distance with increasing code number (j) has to be realized. $L^{(2)} \leq L^{(1)}$ can be obtained by defining $I^{(2)}$ using at least one $j_h = L^{(1)} - L^{(2)}$. This is necessary as the term $D^{L^{(1)}-1}$ appearing in $G^{(1)}$ has to be expressed by $G^{(2)}$ with the maximum exponent $L^{(2)} - 1$. In most cases, j_h will be chosen to lie within the range $0 \leq j_h \leq L^{(1)} - L^{(2)}$.

Example 1: Let code (1) be given by $G^{(1)} = (1 \quad 1 + D + D^2 \quad 1 + D + D^2)$. Defining

$$I^{(2)} \text{ by } I_1^{(2)} = I^{(1)} \text{and } I_2^{(2)} = D \cdot I^{(1)} \text{ with } k^{(2)} = 2$$

leads to the following construction rules for code (2):

$$\begin{aligned} G_{11}^{(2)} + D \cdot G_{21}^{(2)} &= 1 \\ G_{12}^{(2)} + D \cdot G_{22}^{(2)} &= 1 + D + D^2 \\ G_{13}^{(2)} + D \cdot G_{23}^{(2)} &= 1 + D + D^2 \end{aligned} \quad (5)$$

Possible solutions are:

$$G^{(2)} = \begin{pmatrix} 1 & 1+D & 1 \\ 0 & D & 1+D \end{pmatrix} \quad (6)$$

or

$$G^{(2)} = \begin{pmatrix} 1+D & 1+D & 1 \\ 1 & D & 1+D \end{pmatrix} \quad (7)$$

This method can also be used for constructing code (1) by providing code (2) a priori as shown subsequently.

Example 2: Let code (2) be given by

$$G^{(2)} = \begin{pmatrix} 1+D & 1 & 1 & 1 \\ D & D & 1 & 1+D \\ 0 & 1 & D & 1+D \end{pmatrix} \quad (8)$$

and let $L^{(1)}$ be chosen as $L^{(1)} = 3$. Thus, there has to be at least one $j_h = 0$ and one $j_h = L^{(1)} - L^{(2)} = 1$. As $k^{(2)} = 3$, the third $j_h \in \{0,1\}$ can be chosen freely. A possible setting is

$$I_1^{(2)} = D \cdot I^{(1)}, I_2^{(2)} = I^{(1)}, I_3^{(2)} = D \cdot I^{(1)} \quad (9)$$

In this case, the procedure above leads to only one possible solution for the following n equations:

$$G_i^{(1)} = D \cdot G_{1i}^{(2)} + G_{2i}^{(2)} + D \cdot G_{3i}^{(2)} \quad (10)$$

with $i = 1, \dots, n$ and $n = 4$. The Forney matrix of code (1) results in

$$G^{(1)} = (D^2 \quad D \quad 1 + D + D^2 \quad 1 + D + D^2) \quad (11)$$

Note that code (2) cannot be chosen freely, as the resulting code (1) has to consist only of odd weighted generators. Thus, a necessary condition for code (2) is that its generators have odd weight. This is due to the fact that a linear combination of rows of $G^{(2)}$ corresponds to $G^{(1)}$. Such a linear combination can only reduce the number of taps of a generator (column) by an even number. Therefore, odd weighted generators are mandatory even for the second code.

Up to now, this method has only been described for $k^{(1)} = 1$. It can be extended to $k^{(1)} \geq 1$ by the construction

$$I_i^{(2)} = \sum_{h=1}^{k^{(1)}} f_{hi} \cdot I_h^{(1)} \cdot D^{j_{hi}} \quad (12)$$

where $i = 1, \dots, k^{(2)}$, $(h = 1, \dots, k^{(1)})$, $j_{hi} \in \{0, \dots, L^{(1)} - 1\}$ (preferably, $j_{hi} \in \{0, \dots, L^{(1)} - L^{(2)}\}$).

f_{hi} is either 0 or 1 for the h-th input sequence $I_h^{(1)}$ in the i-th equation. We have to ensure that for each h there are at least one $j_{hi} = 0$ and one $j_{hi} = L^{(1)} - L^{(2)}$ (assuming the same order for every row in the Forney matrix) in order to guarantee that each term on the right side of Equation 2 can be expressed by its left side. Thus, $L^{(2)} < L^{(1)}$ can only be realized by our special approach, if

(²)Often, in contrast to our definition, the constraint length is defined as the number of contributing binary positions, which means $L^{(j)} \cdot k^{(j)}$.

$$2 \cdot k^{(1)} \leq k^{(2)} \tag{13}$$

as there are $k^{(2)}$ equations and $2 \cdot k^{(1)}$ prescribed values of j_{hi}.

5. RESULTS OF A COMPUTER SEARCH

The basis for constructing convolutional codes for rotationally invariant encoded QAM are codes with generators of only odd weight. Then, as has been described in section 4, code (2) is constructed from code (1) or vice versa.

Table 1 compiles binary linear convolutional codes with only odd weighted generators with maximum free distance. The list depicts only one example per set of parameters; of course, several different codes have been found.

Table 1 - *Best binary convolutional codes with only odd weighted generators.*

Code	k	n	L	d_f	d_{pro}	d_{max}
(4,7)	1	2	3	4	5	5
(15,16)			4	6	6	6
(32,37)			5	6	7	8
(73,75)			6	8	8	8
(1,2,2)	1	3	2	3	-	-
(4,7,7)			3	6	8	8
(13,15,15)			4	9	10	10
(25,31,37)			5	11	12	12
(51,73,75)			6	13	13	13
(1,2,7,7)	1	4	3	8	10	10
(13,15,16,16)			4	12	13	15
(31,32,32,37)			5	14	16	16
(51,73,73,75)			6	18	18	18
(2,4,7,7,7)	1	5	3	10	13	13
(13,15,15,16,16)			4	15	16	16
(25,32,32,37,37)			5	18	20	20
(32,51,73,73,75)			6	21	22	22
(10,13,15)	2	3	2	3	3	4
(51,61,73)			3	5	5	6
(13,15,15,16)	2	4	2	4	-	-
(37,45,51,76)			3	8	-	-
(13,13,15,16,16)	2	5	2	5	6	6
(46,52,61,67,75)			3	9	10	10
(46,52,61,73)	3	4	2	4	4	4
(46,54,62,73,75)	3	5	2	5	5	5

The notation is chosen according to Proakis [10].(3) d_f is the free distance of the particular code. d_{pro} is the maximum free distance of codes with the same parameters k, n, L (not necessarily generating the all-ones code sequence) published by Proakis [10]. d_{max} is the upperbound on the free distance calculated by Daut [11].

The multilevel construction is derived from codes out of this table. Code examples for the levels one and two of the multilevel scheme are given subsequently in Table 2.

Table 2 - *Codes for M-QAM (Because of the Euclidean distances in the set partitioning scheme and (1), $d_f^{(1)} \geq 2 \cdot d_f^{(2)}$ is aimed at.)*

Code (1)	$R^{(1)}$	$L^{(1)}$	$d_f^{(1)}$	Code(2)	$R^{(2)}$	$L^{(2)}$	$d_f^{(2)}$	Coding gain (dB)
(4,7,7)	$\frac{1}{3}$	3	6	(10,13,15)	$\frac{2}{3}$	2	3	4.7
				(16,13,15)			3	4.7
(15,15,13)		4	9	(51,61,73)		3	5	6.5
(1,2,7,7)	$\frac{1}{4}$	3	8	(46,52,61,73)	$\frac{3}{4}$	2	4	6.0

The coding gain was calculated according to

$$\text{Coding gain (dB)} = 10 \cdot \log_{10} \frac{d_{E_{min}}}{d_{E_{uncod}}} \tag{14}$$

$d_{E_{uncod}}$ is the squared Euclidean distance in the uncoded case with the same average symbol energy. $d_{E_{min}}$ follows from Equation (1) with $d_H^{(j)} = d_f^{(j)}$, the free distance of the j-th level code.

The coding of the levels three, four, and so on, can be chosen freely. E.g., for the level three of the last scheme, a simple parity-check code ($d_H = 2$) will suffice. Of course the redundancy of a parity-check code of length n_P has to be taken into account by a rate loss of

$$-\log_{10}\left(1 - \frac{1}{(ld(M)-1) \cdot n_p}\right)$$

for M-QAM, which is 0.3779 dB for $n_p = 4$ and 0.1848 dB for $n_p = 8$ ($M = 16$). As the loss is relatively low, only the gain concerning the first two levels has been given.

Notice the low complexity of the proposed coding schemes, e.g., 6 dB asymptotic gain with 4 states for the first code and 8 states for the second code.

6. SOME RESULTS FOR $2\pi/M$ - INVARIANT M-PSK

If the third and following partition levels remain uncoded and the differential en/decoders are modified by replacing the modulo-4 adder (subtractor) by modulo - M

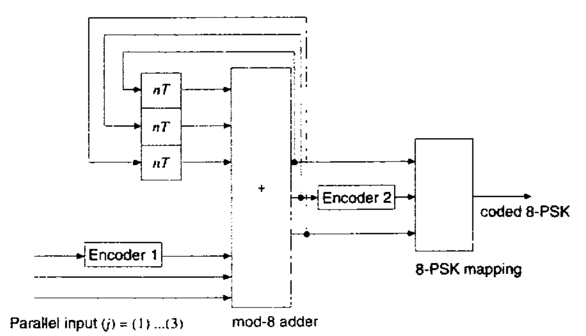

Fig. 6 - Differential encoding for multilevel encoded 8-PSK for 3 dB or unequal error protection 6/3 dB codes (no encoder (3)).

(3) The octal numbers signify the links between the shift register and the modulo-2 adder. The LSB is on the right.

Table 3 - *Codes for* 8 -PSK *(Because of the Euclidean distances in the set partitioning scheme and* (1), $d_f^{(1)} \geq 4 \cdot d_f^{(2)}$ *is aimed at).*

$R^{(1)}$	Code (1)	$L^{(1)}$	$d_f^{(1)}$	$R^{(2)}$	Code (2)	$L^{(2)}$	$d_f^{(2)}$	Coding gain/dB
$\frac{1}{4}$	(4,4,7,7)	3	8	$\frac{3}{4}$	even parity check		2	3.0
	(34,45,75,73)	6	16		(13,25,37,76)	2	4	6.0

operations (Figs. 6, 7), unequal error protection codes can be designed. In the case of coded 8-PSK, we obtain codes with asymptotic coding gains of constant 3 dB or unequal error protection codes with half of the information protected with 6 dB, the remaining half with 3 dB. Two schemes are given in Table 3.

If codes for subsequent levels (beginning with the third) would be introduced, the code conditions will become more difficult [2]. This case has not yet been studied thoroughly.

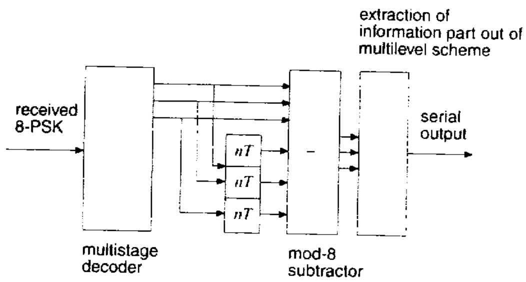

Fig. 7 - Differential decoding for multilevel encoded 8-PSK.

7. CONCLUSIONS

After defining multilevel convolutional codes, we derived conditions for 90° - rotational invariance as was done for the block-coded case in an earlier publication. 90° - invariance is achieved, if the all-ones sequence is valid for the first code and if all sequences of the first code are valid code sequences of the second code as well. We proposed a special differential en/decoding to ensure rotational invariance not only for the code as such, but also for the information sequence. No further conditions were imposed by the differential operations.

When applying the invariance conditions to binary convolutional codes, we found that first all generators need to have odd weight to fulfil the all-ones condition and be non-catastrophic. The second condition was met by a construction, where a special assumption was made on the information sequences of the second component code. They were appointed to be delayed versions of the information sequence(s) of the first code.

Asymptotic coding gains of around 6 dB for M - QAM with a complexity of only 4 states for the first code and 8 states for the second code were obtained.

The same design principles proved to be applicable to coded M -PSK, if only the first two levels are encoded, the others remaining uncoded. Differential en/decoders have also been presented for this case and two examples with asymptotic coding gains of 3 dB and 6/3 dB (unequal error protection) were given.

Further investigations will focus on schemes for 45° - invariant 8-PSK, where also the third level is encoded. Then, the corresponding code conditions become more stringent [2].

Acknowledgements

We thank the referees for their thorough reading, valuable remarks and their obvious interest in the subject. The desired changes have been incorporated as far as possible. We apologize that we cannot include simulation results for the examples as has been demanded by one of the referees. However, it should be noted that, if necessary, there are several possibilities to improve the decoder performance at finite bit-error probabilities: recursive decoding, interleaving between different stages and maybe also soft-output decoding can be applied. See [12] for such proposals.

This paper has in parts also been published at the International Winter Meeting on Coding and Information Theory in Essen, Germany, December 15-17, 1991 and the IEEE International Symposium on Information Theory in San Antonio, TX, USA, January 17-22, 1993.

Manuscript received on November 25, 1992

REFERENCES

[1] T. Kasami, T. Takata, T. Fujiwara, S. Lin: *On linear structure and phase rotation invariant properties of block M -PSK modulation codes.* "IEEE Trans. on Information Theory", Vol. IT-37, No. 1, January 1991, p. 164-167.

[2] W. Henkel: *Phase-invariant coded phase shift keying using Reed-Muller codes.* "AEÜ", Vol. 46, No. 3, 1992, p.125-130. (Minor correction in No. 4).

[3] W. Henkel: *Measures to counteract the influence of phase instabilities on coded-modulation systems.* ICDSC-9, Copenhagen, 18-22 May 92.

[4] W. Henkel: *Conditions for 90° - phase invariant block-coded QAM.* To be published in "IEEE Trans. on Communications".

[5] T. Woerz, J. Hagenauer: *Multistage coding and decoding for a M-PSK system.* Proc. GLOBECOM '90, San Diego, December 2-5, 1990, p. 501.1.1-501.1.6.

[6] G. D. Forney: *Convolutional codes I: algebraic structure*. "IEEE Trans. on Information Theory", Vol. IT-16, No. 6 November 1970, p. 720-738. (Correction May 1971, p. 360).

[7] W. Heise, P. Quattrocchi: *Informations- und codierungstheorie*. 2nd Ed., Springer, Berlin, Heidelberg, New York, 1989, (in German).

[8] E. A. Lee, D. G. Messerschmitt: *Digital communication*. Kluwer, Boston, Dordrecht, London, 1988.

[9] Y. Saito: *Trellis coded modulation for multi-state QAM — which is better, signal space expansion or bandwidth expansion?* "The Trans. of the IEICE", Vol. E 73, No. 10, October 1990, p. 1666-1672.

[10] J. G. Proakis: *Digital communications*. 2nd Ed., McGraw-Hill, Singapore, 1989.

[11] D. G. Daut, J. W. Modestino, L. D. Wismer: *New short constraint length convolutional code construction for selected rational rates*. "IEEE Trans. on Information Theory", Vol. IT-28, No. 5, September 1982, p. 794-800.

[12] T. Woerz, J. Hagenauer: *Iterative decoding for multilevel codes using reliability information*. Subm. GLOBECOM '92.

W. Henkel, M. Koch: **90° - Rotationally Invariant Multilevel Convolutionally Encoded QAM**

ETT. Vol. 4, No. 2 March - April 1993. p. 141 - 147