On Coded M-ary Frequency Shift Keying

A. J. Han Vinck', Alina Hasbi', H.C. Ferreira" and T. Swarts"

' University of Duisburg-Essen
E-mail: Vinck@exp-math.uni-essen.de

"Rand Afrikaanse Universiteit, Johannesburg, South Africa

Abstract

We discuss the combination of permutation codes and M-ary frequency shift keying. We show that this combination has special properties that can be used in power-line communications: e.g. constant envelopes, and error correction capability for impulsive- and narrow band noise. We give a simple convolutional encoding and decoding scheme. The outputs of a binary convolutional codes are mapped onto code words of a permutation code in such a way that the Hamming Distance is preserved, or even increased in some special cases. The demodulator consists of a classical envelope detector followed by a "soft-decision" matrix indicating the order of envelope magnitudes. We also show how permutation codes can be used to allow for multiple users with controlled interference.

1. Introduction

Our research is motivated by the CENELEC regulations for power-line communications in the low frequency region. These regulations require a maximum modulated signal amplitude and thus reduce the number of modulation schemes that can be used. Orthogonal Frequency Division Multiplexing (OFDM) has the disadvantage of a non constant envelope and thus special peak reduction techniques have to be applied. Our approach is to use a constant envelope modulated signal and use coding to combat the noisy environment. In [1] we introduced the combination of M-ary Frequency Shift Keying (M-FSK) and permutation coding. We showed for instance, how the class of permutation codes can be used to correct broad-, narrowband and background noise errors. In section 2 we give a possible receiver structure and the definition of permutation codes. We extend the definition to convolutional codes.

2. Coding and Modulation

2.1. Permutation code and receiver structure

We start with de definition of a permutation code.

<u>Definition 1</u>. A permutation code C consists of |C| code words of length M, where every code word contains the M different integers 1, 2, …, M as symbols.

Every symbol corresponds uniquely to a frequency from an M-ary frequency shift keying modulator, see [1].

<u>Example:</u> For M = 4, a code C with minimum distance 4 has 4 code words: (1, 2, 3, 4), (2, 3, 4, 1), (3, 4, 1, 2), (3, 1, 2, 3).

We can upper bound the number of code words by:

$$|C| \leq \frac{M!}{(d_{\min}-1)!}.$$

Results on constructions can be found in [2, 5]. We can decode the code words with the following detector structure.

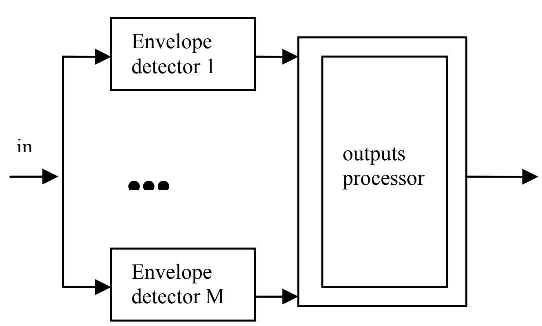

Fig. 1. General receiver structure

The processor operates on the envelope detector outputs. When the channel conditions are known exactly, we may use a maximum likelihood (ML) processor and find the ML code word from the observed envelope detector outputs. However, we assume that the Power-line channel is not exactly known and opt for "sub-optimal" detection. As an example one could opt for a "soft-output" processor, which is best explained using the following example.

Example. For M = 4, after the processor ranks the envelope detector outputs, i.e. we obtain a 4 x 4 matrix containing the following elements:

1	3	4	2
2	1	3	4
4	2	2	1
3	4	1	3

The value 3 at row 1 and column 2 indicates that the envelope for the frequency 1 is the 3rd largest of the 4.
The decoder for this situation now searches for the code word with the minimum overall weight.
For M = 4 and minimum distance 4, we expect a gain of about 3 db, see also the simulation results in Fig. 2.

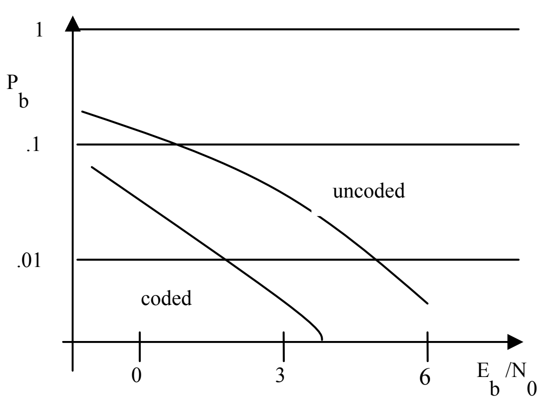

Fig. 2. Simulation results for M = 4 and "soft-decision"

We could also use an alternative procedure with thresholds, i.e. whenever an envelope is above the threshold for that particular frequency we output a 1, otherwise a zero. The 4 x 4 matrix then might look like

1	0	0	1
1	1	0	0
0	0	1	1
0	0	1	0

This is a kind of hard decision version of the previous method. The advantage is that complicated weighting can be avoided. The disadvantage of the "hard-decision" is that we need to set the thresholds. The threshold setting depends on the received signal energy per channel and the probability density function of the noise. The fact that every code word consists of 4 different symbols can be used to determine the necessary values. The decoder searches for the code word with the maximum number of matches.
Channel noise causes errors in the matrix. We have the following type of errors:

- background noise: insertion or deletion of ones
- impulsive noise: a complete column is set to 1
- narrow band noise: a row is set to 1

For a code with minimum distance d_{min}, d_{min} -1 of the above type of error events still lead to correct decoding of the permutation code word. The reason for this is that the above error events agree with a code word from the permutation code in only one position. Since code words differ in at least d_{min} positions, less than $d_{min} - 1$ errors can never give rise to another code word.
Iterative decoding schemes can be found in [4], where the authors develop iterative decoding for OFDM (which is a more complex modulation technique than M-FSK) for impulsive noise channels.

2.2. Extension to Convolutional Codes

Since block code constructions and decoding is getting complicated for larger values of M, we extend the idea of permutation codes to convolutional codes. These codes can be decoded using a trellis and obtain a high distance, call free distance at a high code efficiency. The concept is given in Fig. 3.

For every n convolutional encoder output symbols the mapping outputs a code word from a permutation code. The key idea is to find an ordered subset of 2 M-tuples, out of the full set of permutation M-tuples (with cardinality M!), such that the Hamming distance between any two permutation M-tuples is at least as large as the distance between the corresponding convolutional code's output n-tuples which are mapped onto them.

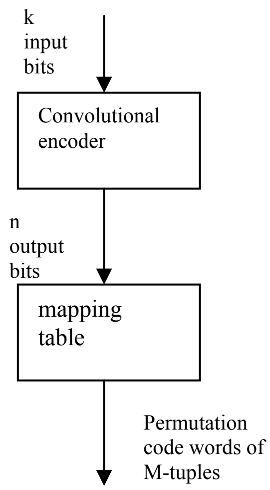

Fig. 3. Permutation convolutional encoding.

Example: For M = 3 and a k/n = ½ convolutional encoder we have the mapping table 1.

Table 1
Mapping from convolutional output to permutation code word

Convolutional Output	permutation code word
0 0	2 3 1
0 1	2 1 3
1 0	1 3 2
1 1	1 2 3

For the mapping in Table 1 we observe that the distance between any two 2-tuples is increased by one for the permutation codes words. As an example, the distance between 01 and 10 is 2 and the corresponding distance between 213 and 132 is 3. For a simple convolutional encoder with memory 2, we obtain a "free distance" of 8 for the permutation trellis and 5 for the binary trellis. We call this a distance increased mapping. More details can be found in [3]. The Viterbi algorithm can now be used to decode the ML path in the trellis of permutation code words.

Figure 4 gives the simulation results for impulsive noise and R=1/4 (M=4), R=1/2 (M=3), R = 4/5 (M=5) convolutional code with free distance 8, 5 and 2, respectively. The distance between the n tuples was the same as the distance between the permutation code words. We call this distance conservative mapping.

The same type of simulation results are obtained for narrow band noise and background noise.

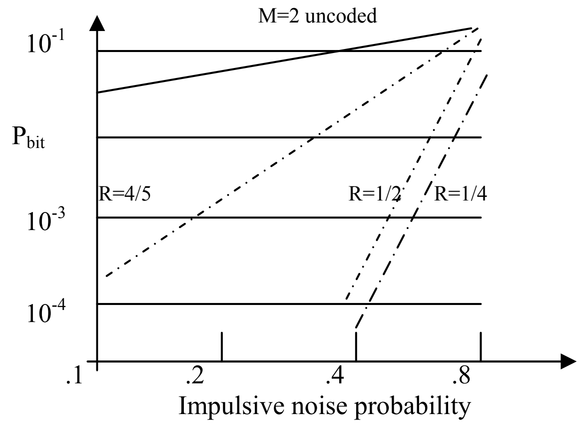

Fig. 4. Bit error rate for impulsive noise.

3. Multi User Codes

In this section we indicate how permutation codes can be used as multi user signaling codes. We distinguish between the situations where the active users transmit one bit of information using a signature and the situation where the users transmit $\log_2 M$ bits of information by selecting a particular sequence out of M private signatures.

3.1. Binary transmission.

Suppose that we have a permutation code of length M with minimum distance d_{min}. Every user gets assigned a particular code word (signature) and a common code word. The number of agreements between any two signatures is less than M-d_{min}. The active users transmit their signature for a 1 and the common word for a 0. For A active users, no other code word is generated when A*(M-d_{min}) < M, and thus unique decoding can be achieved. The sum rate of the system for A active users is

$$R_{sum} = \frac{A}{M^2} < \frac{M}{M - d_{min}} \times \frac{1}{M^2} = \frac{1}{(M - d_{min})M}.$$

As an example, for M = 5 and d_{min} = 4, |C| = 120. For A < 5, we have the sum rate R_{sum} < 1/5. For TDMA the sum rate would be less than $5\log_2 5/120$.

3.2. M-ary transmission

In this case, we give every user M different code words from a permutation code with M(M-1) code words and minimum distance M-1. The maximum number of users is M-1 and the maximum interference is thus equal to 1. This means that up to M-1 users can transmit simultaneously and still unique detection is possible, because one unique position is left.

If we shorten the code words to length n, the minimum distance is reduced to n-1, and thus the maximum interference is equal to 1. Note that the code word is a word from a permutation code with minimum distance n-1 and length n. For A < n active users, we still have unique decoding. The normalized sum rate is given by

$$R_{sum} = \frac{A}{Mn}\log_2 M \le \frac{n-1}{n} \times \frac{\log_2 M}{M} \Rightarrow \frac{\log_2 M}{M},$$

which is the same as single user transmission.

For A ≥ n, we have a detection error when the active users generate a valid code word. For n ≤ A ≤ M, the error probability can be upper bounded by

$$P_e \le (M-1)[\left(\frac{1}{M}\right)^n \binom{A-1}{n} n!] \approx M\left(\frac{A}{M}\right)^n.$$

For TDMA, the sum rate

$$R_{sum}(TDMA) = \frac{A}{M^2}\log_2 M \le \frac{\log_2 M}{M}.$$

For M a prime number, the encoding can be done with the following matrix operation, where m, 0 ≤ m ≤ M-1 is a message value, I the user number, 0 < I < M, n ≤ M and all operations modulo M. The code word for message m and user I is given by

$$C_I(m) = (m, I)\begin{pmatrix} 1 & 1 & \bullet\bullet\bullet & 1 \\ 0 & 1 & \bullet\bullet\bullet & n-1 \end{pmatrix}.$$

<u>Example.</u> M = 5, n = 3, user 2 and message 3.

$$(3,0,2) = (3,2)\begin{pmatrix} 1 & 1 & 1 \\ 0 & 1 & 2 \end{pmatrix}.$$

Figure 5 gives the performance of a particular example.

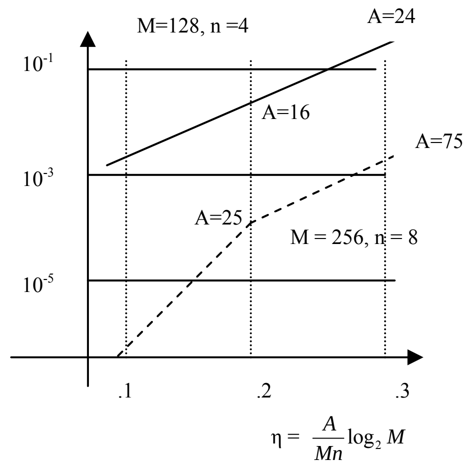

Fig. 5. Performance for M = 128 and M = 256

3.3. "Group" time sharing

In a simple time sharing system, the maximum normalized rate is $R_{TDMA} = \log_2 M/M$. However, by using "group" time sharing, the maximum capacity $R_{Group\text{-}TDMA}$ = (M-1) bits per channel use can be obtained. The "group" time sharing operates as follows. Divide the users into groups of order M-1. User I, 0 < I < M, in the group gets two frequencies, the basis frequency f_0 and the frequency f_I. The basis frequency f_0 is common for all users in the system. When it is time for a particular group to transmit its information, the users in the group transmit f_0 for an information bit 0, and their particular assigned frequency for a bit 1. The receiver detects the transmitted frequencies and is capable of uniquely detecting the information for all involved users. Since we have M-1 users in a group, we transmit (M-1) bits of information per channel use.

Example. M = 4.

Information bit	0	1
User I	f_0	f_I
User I+1	f_0	f_{I+1}
User I+2	f_0	f_{I+2}

Possible detector output:

(f_0)
(f_0 f_I), (f_0 f_{I+1}), (f_0 f_{I+2})
(f_0 f_I f_{I+2}), (f_0 f_{I+1} f_{I+2}), (f_0 f_I f_{I+1})
(f_I f_{I+1} f_{I+2})

4. Conclusions

We showed that Permutation codes are attractive alternatives in a power line communication environments. We did not give a real implementation, but indicated construction methods and application in a multi user environment. It is a challenge for the future to realize a particular scheme.

Acknowledgements

The first author wants to express his gratitude to the Minister of Education of the Taiwanese Government for making a research visit to Taiwan possible. He also wants to thank Professor Hswa-Hwa Chen from the Sun Yat-sen University in Kaohsiung, Taiwan, for his support.

References

[1] A.J. Han Vinck, “Coded Modulation for Powerline Communications,” AEU International Journal of Electronics and Communications, Vol. 54, No. 1, 2000, pp. 45-49.

[2] I.F. Blake, “Permutation Codes for Discrete Channels,” IEEE Tr. on Information Theory, Vol. 20, Jan. 1974, pp. 138-140.

[3] H. Ferreira and A.J. Han Vinck, ”Interference cancellation with Permutation Trellis Codes,” Proceedings of the IEEE Veh. Techn. Conference Fall 2000, September 24-28, 2000, Boston, MA, USA, pp. 4.5.1.3.1 B 4.5.1.3.7.

[4] Juergen Haering and A.J. Han Vinck, “Iterative Decoding of Codes over Complex Numbers for Impulsive Noise Channels,” IEEE Transactions on Information theory, May 2003, pp. 1251-1260.

[5] Tadashi Wadayama and A.J. Han Vinck, “A Multilevel Construction of Permutation Codes,” IEICE Transactions on Fundamentals, October 2001, pp. 2518-2522.