Interleaved Block Codes for the Photon Channel

R. J. McEliece
Communications Systems Research Section

Using a recent idea of J. Massey, we show that interleaved binary block codes combined with pulse position modulation give the best practical coded systems yet devised for optical communication with photon detection.

I. Introduction

In Ref. 1, the use of Reed-Solomon (RS) codes with pulse-position modulation (PPM) was suggested for optical communication using direct photon detection. In Ref. 2, it was shown that PPM is optimal or nearly so in this application, but with no guarantee that RS coding cannot be improved upon. Indeed, in a recent article Massey (Ref. 3) has suggested the use of interleaved binary convolutional codes for this application, and these codes perform almost as well as RS codes. In this article we shall expand on Massey's suggestion and show that interleaved binary block codes can, for a given decoder complexity, perform even better than RS codes. In Fig. 1 we will present performance curves for an explicit code with a bit error probability less than 10^{-6} at a code efficiency of 2.8 nats per photon, using 256-ary PPM. Since channel capacity with this level of PPM is only 5.6 nats per photon, this performance is about as good as could be hoped for with any reasonable decoder complexity.

II. Massey's Equivalence

When M-ary PPM is used, the photon channel becomes, as explained in Ref. 1, an M-ary erasure channel with erasure probability $\epsilon = e^{-\lambda}$, where λ is the expected number of photons received during a time slot when the transmitter is pulsed. How should we code for this channel? In Ref. 1 it was shown that if $M = 2^L$, Reed-Solomon codes over $GF(2^L)$ give good results. More recently Massey (Ref. 3) has suggested that when $M = 2^L$, it might be wise to view the M-ary erasure channel as an array of L parallel, completely correlated, binary erasure channels. It is this latter possibility that we wish to explore here.

Massey's idea is simply that when $M = 2^L$, each input letter to the M-ary erasure channel can be represented by L bits. When this input letter is received correctly, all L bits are received correctly. However, if this input letter is erased, all L bits are erased. For example if $L = 3$, and if $\{0,1,2,3,4,5,6,7\}$ is the transmission alphabet, the sequence 314152653 might be received as 314?5?65?, where "?" denotes a channel erasure. Using the standard octal code 0 = 000, . . . , 7 = 111, this sequence could be viewed as three parallel binary sequences, as follows:

8-ary stream	3 1 4 1 5 2 6 5 3	→	3 1 4 ? 5 ? 6 5 ?
3 parallel binary streams	0 0 1 0 1 0 1 1 0 1 0 0 0 0 1 1 0 1 1 1 0 1 1 0 0 1 1	→	0 0 1 ? 1 ? 1 1 ? 1 0 0 ? 0 ? 1 0 ? 1 1 0 ? 1 ? 0 1 ?
	transmitted		received

Notice that the erasures occuring in the L parallel binary channels occur in exactly the same locations, i.e., if the k-th transmitted bit is erased in any one of the channels, it will be erased in all of them. Thus the 2^L-ary erasure channel is equivalent to L parallel, completely correlated, binary erasure channels as Massey observed. In Ref. 3, Massey suggests that, in view of this equivalence, it might be worthwhile to code for this channel by using an L-fold interleaving of a good code for a single binary erasure channel (BEC). The codes he suggests for this use are in fact short constraint-length convolutional codes with Viterbi decoders. In this paper we will also investigate this interleaving idea, but will consider linear block codes rather than convolutional codes.

III. Linear Block Codes for the BEC

It has been known since the mid-1950's (Ref. 4) that linear block codes are especially well-suited for combatting erasures. Rather than give an abstract explanation of this fact, we will illustrate it by example. We will also consider the implementation of interleaved linear codes on the parallel erasure channels discussed in the last section.

Consider the (8,4) $d = 4$ extended Hamming code with parity-check matrix

$$\begin{matrix} & 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 \end{matrix} \quad \text{column indices}$$

$$H = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Suppose we receive the word [1?0???01]. How should we decode it? The idea is to try to express the erased coordinates in terms of the unerased coordinates. In the present case, coordinates 1,3,4,5 have been erased, and to decode we reorder the columns of H so that the columns corresponding to erased coordinates all appear on the left:

$$\begin{matrix} & 1 & 3 & 4 & 5 & 0 & 2 & 6 & 7 \end{matrix}$$

$$H = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \end{bmatrix}$$

Next, using elementary row operations, we put H into row-reduced echelon form H'. Omitting details, we obtain the result

$$\begin{matrix} & 1 & 3 & 4 & 5 & 0 & 2 & 5 & 7 \end{matrix}$$

$$H' = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 \end{bmatrix}$$

This matrix expresses the erased coordinates as linear combinations of the unerased coordinates, as desired. This is because the definition of a codeword X is the equation $HX^T = 0$, which is equivalent to $H'X^T = 0$, which in our example is equivalent to the four equations:

$$X_1 = X_0 + X_6 + X_7$$

$$X_3 = X_0 + X_2 + X_6$$

$$X_4 = X_2 + X_6 + X_7$$

$$X_5 = X_0 + X_2 + X_7$$

Hence the word [1?0???01] with $X_0 = 1$, $X_2 = 0$, $X_6 = 0$, $X_7 = 1$, must be a garbled version of the codeword [10011001], and the decoding is complete.

Now this particular code is not capable of correcting all patterns of four erasures; e.g., if we had received [1?0??0?1], we would compute

$$\begin{matrix} & 1\,3\,4\,6\,0\,2\,5\,7 \end{matrix} \qquad \begin{matrix} & 1\,3\,4\,6\,0\,2\,5\,7 \end{matrix}$$

$$H = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \end{bmatrix}, H' = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}$$

In this case H' tells us that the erased coordinates X_1, X_3, X_4 can be expressed in terms of the unerased coordinates X_0, X_2, X_5, X_7 plus the erased position X_6:

$$X_1 = X_6 + X_2 + X_5$$

$$X_3 = X_6 + X_0 + X_2$$

$$X_4 = X_6 + X_0 + X_5$$

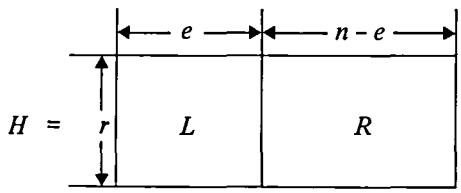
Since we do not know the value of X_6, there are in this case *two* possibilities for the transmitted codeword:

$$X_6 = 0: \ [10011001]$$

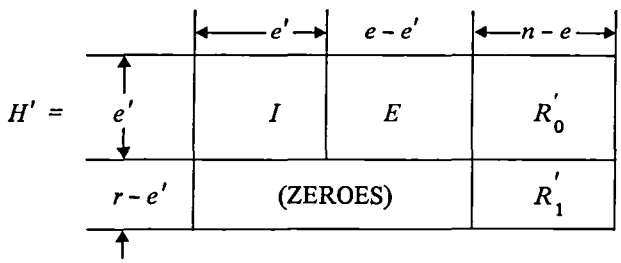
$$X_6 = 1: \ [11000011]$$

These are both bona fide codewords, and both agree with the received word on all four unerased positions. In this case our decoder fails, but in any case detects its own failure.

The general situation is this. H will be an $r \times n$ binary matrix, where $r = n - k$ is the code's redundancy. If e positions are erased by the channel, after reordering, the columns H will have the form



After row-reduction, H' will look like this:



where $e' \leqslant e$, and I is an $e' \times e'$ identity matrix. The given erasure pattern will be correctable if and only if $e' = e$, in which case the top e rows of H' will express the e erased coordinates in terms of the $n - e$ unerased coordinates. (The lower right-hand matrix R_1', expresses parity-checks that must be satisfied by the unerased positions. In the present application this is not useful, but if bit errors as well as erasures are present, R_1' can be used to help locate the errors.) The amount of computation required to row-reduce H is at most r^2 row operations, or $r^2 n$ bit operations. Once H' is known, it requires at most r further row operations, or rn bit operations, to recover the erased coordinates. Since each codeword carries $k = n - r$ information bits, the total computational effort is at most $A(r + 1)$ row operations, or $A(rn + n)$ bit operations per decoded bit, where $A = (1 - R)/R$ and $R = k/n$ is the code's rate.

If we want to use L interleaved copies of this code on the L parallel, completely correlated BECs described in the last section, the decoding effort per bit is considerably reduced. This is because all L garbled codewords will have the same erased positions, and so the reduction from H to H' need only be done once. Thus decoding L codewords requires at most $r^2 + Lr$ row operations or at most $A/L\ r + A$ row operations, or $A/L\ rn + An$ bit operations, per decoded bit. For most choices for the parameters, this is very nearly a savings of a factor of L.

In the next section we will show how these results can be used to design good coding systems for the photon channel.

IV. Performance of Interleaved Block Codes on the Photon Channel

In the last section we discussed the decoding of linear block codes on a BEC, but did not discuss the performance of these codes. For a given choice of n and r, it is in general not easy to find the $r \times n$ parity check matrix H that describes the best possible erasure-correcting linear block code. However, we can give conservative estimates of the performance of linear codes by using the "R_0-coding theorem" (Refs. 4 and 5) which says in this case that for a given choice of r and n, a randomly chosen $r \times n$ parity check matrix H yields a code with probability of decoding error bounded by

$$P_E \leqslant \frac{(1 + \epsilon)^n}{2^r}, \qquad (1)$$

where ϵ is the channel's erasure probability. If a code with these parameters is used on the L parallel erasure channels corresponding to the photon channel with 2^L-ary PPM, the erasure probability ϵ is given by Ref. 1 as

$$\epsilon = 2^{-LR/\rho}, \qquad (2)$$

where $R = (n - r)/n$ is the code's rate, and ρ is the code's efficiency measured in nats per photon. Combining Eqs. (1) and (2) we have plotted in Fig. 1 (conservative estimates of) the bit error probability of randomly selected (100,50), (200,100), and (300,100) linear codes interleaved to depths $L = 5$ and $L = 8$. Presumably, carefully selected codes with

these parameters would perform somewhat better. The computational effort per decoded bit is, from the remark in the last section, seen to be:

21 row operations (100,50) $L = 5$

14 row operations (100,50) $L = 8$

41 row operations (200,100) $L = 5$

26 row operations (200,100) $L = 8$

121 row operations (300,100) $L = 5$

76 row operations (300,100) $L = 8$.

We note from Fig. 1 that the (300,100) $L = 8$ linear code slightly outperforms the RS code suggested in Ref. 1, and a vector-oriented special purpose decoder for this code would be a considerably simpler device than the corresponding RS decoder.

References

1. McEliece, R. J. and L. R. Welch, "Coding for Optical Channels with Photon Counting," *DSN Progress Report 42-52* (1979), pp. 61-66.
2. McEliece, R. J., Rodemich, E. R., and Rubin, A. L., "The Practical Limits of Photon Communications," *DSN Progress Report 42-55* (1979), pp. 63-67.
3. Massey, J., "Capacity, Cut-Off Rate, and Coding for a Direct-Detection Optical Channel," article in press.
4. Berlekamp, E. R., "Error-Correcting Codes," *Proceedings of the IEEE*, May 1980, pp. 564-593.
5. McEliece, R. J. *The Theory of Information and Coding*, Reading Mass., Addison-Wesley, 1977.

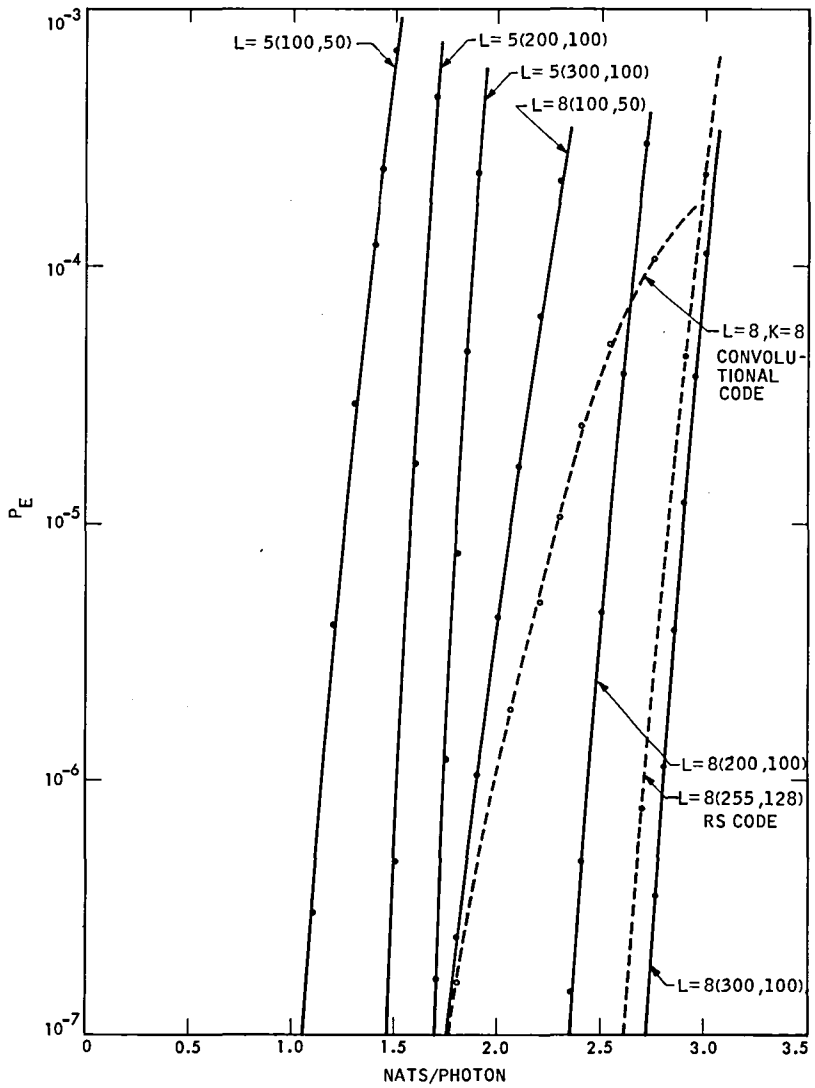


Fig. 1. Code performance curves