Codes for Second and Third Order GH-ARQ Schemes

Luca de Alfaro and Angelo Raffaele Meo

Abstract— **In the usual ARQ (Automatic-Repeat-Request) communication protocols, when the receiver detects the presence of errors in a received message, it requests the retransmission of another copy of the message, and the process continues until an uncorrupted copy reaches the destination. If the quality of the channel is poor, the number of retransmissions may become very large.**

In order to improve the efficiency at large channel error rates, some new techniques known as Type-II Hybrid ARQ Schemes have recently been proposed. In these techniques the transmitter does not retransmit the message encoded as in the first transmission: it sends instead additional redundancy that is decoded at the destination along with the previously received information to allow error correction.

In this paper a new kind of Type-II Hybrid ARQ transmission scheme is proposed. The copies of the message are encoded in different ways, so that each copy has the same information content, and the knowledge of any subset of the copies allows error correction. This transmission scheme is well suited to broadcast as well as for point to point communications.

We have obtained trough computer search a family of optimum and quasi-optimum codes for the proposed transmission scheme. They are described along with new decoding techniques of remarkable simplicity and effectiveness. These codes allow a message to be encoded in up to three different ways, and they are shown to be simple to encode and decode and to perform well over channels affected by error bursts.

I. Introduction

Most schemes of data transmission across a point-to-point link use a mechanism of error control known as ARQ (Automatic-Repeat-Request). When the controlling device at the destination detects one or more errors in the received message, it requests a new copy of the same message to be sent. The same frame is sent again using the same code, and the procedure is repeated until the destination controlling device accepts the message as error-free.

The same scheme is also applied to the broadcast from a single source to a multitude of receivers as in teletext. In this case the same messages are transmitted repeatedly, according to a specific cyclic order. When the controlling device of the receiver detects one or more errors in a received message, it waits for the next transmission of the same message, that will be accomplished using the same code.

Paper approved by Ezio Biglieri, the Editor for Data Communications and Modulation of the IEEE Communications Society. Manuscript received January 2, 1991; revised July 7, 1991 and February 7, 1992.
L. de Alfaro was at Dipartimento di Automatica ed Informatica, Politecnico di Torino, Corso Duca degli Abruzzi 24, 10129 Torino, Italy, and is now at Department of Computer Science, Stanford University, Stanford CA 94305. A.R. Meo is at Centro Elaborazione Numerale dei Segnali, Dipartimento di Automatica ed Informatica, Politecnico di Torino, Corso Duca degli Abruzzi 24, 10129 Torino, Italy.

In both cases, if the quality of the channel is poor, the average number of retransmissions may become very large. Therefore, the throughput efficiency of ARQ systems is a rapidly decreasing function of the channel error rate. This drawback is due to the use of the same code in all transmissions, which makes it difficult to use the information content of the first received frame to correct the new one.

In order to improve both efficiency and reliability even at large channel error rates, some techniques known as "Type-II Hybrid ARQ" have been proposed. These hybrid techniques combine the advantages of high efficiency and throughput regardless of the channel quality of FEC (Forward-Error-Control) schemes and those of high communication reliability of ARQ systems. In the type II hybrid techniques the transmitter, upon receipt of a negative acknowledgment (NAK), does not retransmit the same codewords as on the original transmission, but sends additional redundancy that is decoded at the destination along with the received words stored from one or more previous transmissions.

The hybrid schemes may be seen as an extension of the idea of transmitting the parity-check bits only when they are needed. The idea was first introduced by Mandelbaum [1], who proposed punctured codes for transmitting redundancy in incremental steps by using Reed-Solomon codes. The first extensions of Mandelbaum's idea were described by Metzner [3, 4], Ancheta [5], Lin and Ma [6] and Lin and Yu [7]. Other combined ARQ/FEC schemes have been presented by Lin and Costello [8, 9].

Among the more recent contributions, the technique of combining all the received copies of a packet in order to obtain useful throughput at very high channel error rates appears very interesting [10, 11, 12, 13, 14]. A significant amount of work has been done on hybrid ARQ/FEC schemes using both convolutional coding and sequential decoding [15, 16, 17, 18].

This paper follows the line of research that was opened up by Yu and Lin [19, 20], and by Wang and Lin [21]. This line is characterized by a scheme that uses two linear codes C_0 and C_1 where C_0 is an (n, k) error-detecting code and C_1 is a $(2n, n)$ half-rate invertible code, designed for error correction only. The two half-codewords produced by C_1 are alternatively used in two successive transmissions.

The scheme of Yu, Lin and Wang has recently been extended by Krishna and Morgera [22], who have introduced the concept of a generalized hybrid ARQ (GH-ARQ) scheme for adaptive error control and have proposed a new class of linear codes for such systems. The GH-ARQ scheme uses a high-rate (n, k) code C_0 for error detection only and a second code C_1 for adaptive error correction.

The code C_1 is an (mn, n) error-correcting code having distance d, selected in such a way that its generator matrix G can be partitioned into m sub-blocks each of dimension $n \times n$:

$$G = [G_1|G_2|\ldots|G_m]$$

The minimum distance d_i of the subcode $[G_1|G_2|\ldots|G_i]$ is less than d_j for all $1 \leq i \leq j \leq m$, so that the correction capability increases at any new transmission.

This paper proposes a slightly more general GH-ARQ transmission scheme than the one proposed by Krishna and Morgera. The most important difference is that in a cyclic transmission any two adjacent codewords belong to an optimized code. This feature makes our scheme suitable both to point-to-point communications and to continuous broadcast, used for example by teletext or by the recent datacast, where the receiver may be connected during the transmission of any encoding. However, the proposed codes do not go beyond the third order.

Second, a new error correction technique for third order codes is described. In comparison with traditional minimum distance decoding, this technique offers enhanced performance in the presence of burst errors in addition to lower computational complexity.

Third, several optimum and quasi-optimum linear codes for second and third order GH-ARQ schemes are presented. These codes have been found by using new computation tools that have been specifically designed for this research.

Fourth, the problem of bit and packet synchronization is addressed. This problem is very important when the bit error rate is large. Second- and third-order optimum and quasi-optimum codes are proposed that not only allow error detection and correction, but also take care of maintaining the transmitter and receiver synchronized even in the presence of channel errors.

II. The Transmission Scheme

The proposed transmission scheme is shown in Fig. 1. The source message is subdivided into l_1 blocks, each m bits long, typically with $8 \leq m \leq 16$. The source message is followed by its frame check sequence (FCS), subdivided into l_2 blocks of m bits.

The source message is encoded using two or three different codes, according to the transmission scheme used. These different encodings of the message are transmitted cyclically over the channel until the receiver is able to reconstruct the original message. A message encoded with the first encoding is composed of the following fields:

1. an initial sequence s used for frame synchronization;
2. a message header h containing the number of the encoding used and, possibly, the message number in the transmission window;
3. the sequence of blocks $\overline{v}_{1i} = \varphi(\overline{u}_i)$, $1 \leq i \leq l_1 + l_2$, obtained from the source blocks $\overline{u}_i$ by applying the function φ that implements the synchronization specifications. Usually, every block $\overline{v}_{1i}$ has a length $k > m$;
4. a delimiter sequence d signalling the end of the message.

The second and third encodings are subdivided into the same sequences as the first encoding. However, any information or FCS block $\overline{v}_{2i}$ of the second encoding will be related to the corresponding block $\overline{v}_{1i}$ by the linear relationship $\overline{v}_{2i} = A\overline{v}_{1i}$, $1 \leq i \leq l_1 + l_2$. Similarly, any information or FCS block of the third encoding is given by $\overline{v}_{3i} = B\overline{v}_{2i}$, $1 \leq i \leq l_1 + l_2$, where A and B are two $k \times k$ square matrices specifically chosen in order to improve the error correction capability of the receiver.

The model of Fig. 1 has been oriented to both point-to-point and broadcast communications. This implies that the second or third encoding may be the first to be received and, therefore, that the matrices A, B, and BA must be invertible to allow the recovery of the original message from any of the encodings.

III. Simple Second-Order Schemes

In this section we will consider the case of second order schemes without taking into account the synchronization problems, which will be dealt with in Section V. In this case the scheme shown in Fig. 1 is simplified as follows:

1. Only the first and second encodings are involved.
2. The header h is used to distinguish between the two different possible encodings of the channel message. Since its information content is equivalent to one bit, it can be easily expressed with a high level of reliability in a block of k bits. For example, if a simple repetition code is used and $k = 8$, the bit pattern 00000000 would indicate the first encoding and the pattern 11111111 would indicate the second one.
3. The function φ is the identity function, and therefore $k = m$. Thus, with the only exceptions of s, h and d, the first encoding coincides with the source message followed by its FCS: $\overline{v}_{1i} = \overline{u}_i$; $\overline{v}_{2i} = A\overline{v}_{1i} = A\overline{u}_i$, $1 \leq i \leq l_1 + l_2$.

The code involved in a simple second order scheme is a $(2k, k, d)$ linear invertible code described by its matrix A, where $2k$ is the length of the code, 2^k the number of codewords and d the minimum Hamming distance between two codewords belonging to the code. The code is *optimum* from the point of view of error correction if and only if no other linear invertible code $(2k, k, d')$ exists with $d' > d$.

In order to determine optimum and quasi-optimum codes for second and third order schemes, a new set of computation tools has been developed. In the case of simple second order schemes these tools use the following theorems, whose proof is straightforward and is left to the reader, and which are consequences of the fact that in a linear code no fewer than d columns of the parity check matrix can sum to 0 [2].

Theorem 1: If $H = [A|I_k]$ is the parity check matrix of a $(2k, k, d)$ linear code, then for any $0 < j < d$, the sum

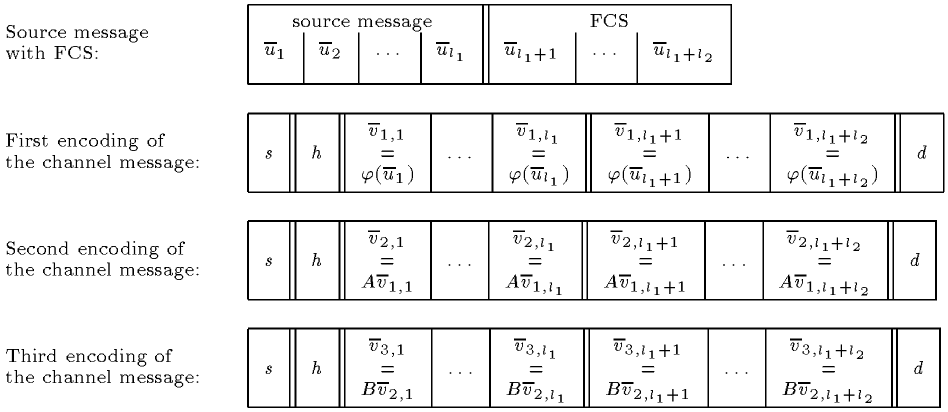

Fig. 1. The cyclic GH-ARQ scheme.

(modulo 2) of j columns of A contains at least $d - j$ elements equal to 1.

Theorem 2: If there exists a d such that, for any $0 < j < d$, the sum (modulo 2) of any j columns of A contains at least $d - j$ elements equal to 1, then the minimum distance of the code described by A is at least d.

The algorithm used to determine the matrix A of a linear invertible code $(2k, k, d)$ builds the matrix one column at a time from left to right, verifying for each new column if it satisfies the following conditions:

1. it is linearly independent from all preceding columns, in order to have $\det A = 1$, as required by the invertibility condition;
2. each of the possible sums of this column with j preceding columns, where $0 < j < d - 1$, contain at least $d - j - 1$ elements equal to 1, as required by Theorem 1.

If the column being considered satisfies conditions 1 and 2, it is placed on the right of those already found, the variable indicating the number of already found columns is increased by one, and the process continues until all k columns have been determined. If at some point all the possible columns have been tested without finding one satisfying conditions 1 and 2, then the number of found columns is decreased by one, the matrix A is shrunk deleting the rightmost column and the search continues.

The algorithm is equivalent to the traversal of a tree whose nodes represent ordered sets of columns that have already been chosen. Every node has as many children as there are valid columns that can be added to the partially determined matrix A associated with the node: a column a_j can be added to a set of already determined columns $a_1, a_2, \ldots, a_{j-1}$ if the above conditions 1 and 2 are satisfied. The solutions are represented by leaves having depth k, meaning that all the columns have been successfully determined.

To speed up the search, some additional tricks have been developed to prune the search tree without losing any solution. However, the time required for the traversal grows exponentially with the dimension k of the matrix, which makes this method impractical for values of k much larger than those reported in this paper.

Some optimum second order codes are shown in Table I.

Table I.

Optimum Codes for Second Order Schemes. The Columns of Matrix A are Represented in Hexadecimal Notation.

(2k,k,d)	A
(8,4,4)	07 0B 0D 0E
(10,5,4)	07 0B 0D 0E 13
(12,6,4)	07 0B 0D 0E 13 23
(14,7,4)	07 0B 0D 0E 13 23 43
(16,8,5)	0F 33 55 6A 96 AB DB ED
(18,9,6)	01F 067 0AB 0D5 12D 156 1B7 1DB 1EE
(20,10,6)	01F 067 0AB 0D5 12D 156 1B7 1DB 1EE 22E
(22,11,7)	03F 1C7 2D9 36A 3B4 4EB 571 59D 67C 727 7D2
(24,12,8)	07F 38F 5B3 6D5 769 9D6 AE3 B3A CF8 D1D E4E FA4

A. The Correction Procedure

We consider here the case of broadcast, since the case of point-to-point communication can be viewed as a simple variant of the former. The transmitter transmits alternatively the first and second encoding of the message, either forever in the case of continuous broadcast, or until the receiver or the receivers have correctly reconstructed the message. In the case of point-to-point communications, the receiver always receives the first encoding first; in the case of continuous broadcast the receiver may begin to listen during the transmission of any encoding. The receiver executes the following procedure:

1. If only the first encoding of the message has so far been received, the receiver verifies the correctness of the transmission by using the FCS. If no errors are detected, the message is accepted. Otherwise, a NAK is

sent to request the transmission of a different encoding of the same message.

2. If only the second encoding of the message has so far been received, the receiver reconstructs the first encoding applying the relation $\overline{u}_i = \overline{v}_{1i} = A^{-1}\overline{v}_{2i}$, $1 \leq i \leq l_1 + l_2$, and then verifies the frame using the FCS of the reconstructed message. If no errors are detected, the message is accepted; otherwise, a NAK is sent to request the transmission of a different encoding of the same message.
3. When a different encoding of the message is received, the receiver first checks if this last transmission has been affected by errors, using the methods explained in steps 1 and 2 for messages encoded with the first and second encoding respectively. If no errors are detected, the message is accepted. Otherwise, the receiver performs the joint correction of the last two different encodings received. Each pair of corresponding blocks $\overline{v}_{1i}$ and $\overline{v}_{2i}$, considered together, forms a codeword of a linear half-rate $(2k, k, d)$ error correcting code, and an estimate of the source block $\overline{u}_i = \overline{v}_{1i}$ can be obtained by using standard decoding technique [8]. The receiver then verifies the correctness of the transmission using the reconstructed FCS. If no errors are detected at this stage, the message is accepted. Otherwise, the receiver requests a new copy of the message to be sent, and the process continues from step 3 until the receiver is able to reconstruct a message which is either error-free or affected by undetectable errors.

IV. Simple Third-Order Schemes

In the third order schemes the source message is encoded in three possible ways. In this section, we will not deal with the synchronization problems, which will be covered in Section V. We will therefore take φ to be the identity function, and $k = m$.

As in the case of second order simplified schemes, the first encoding is equal to the source message followed by its FCS, with the fields h, s and d added. The header h contains the number of the encoding used for the channel message. Since this information is only ternary, it is very easy to transmit it at very high levels of reliability.

The three encodings are related by, for $1 \leq i \leq l_1 + l_2$,

$$\begin{aligned} \overline{v}_{1i} &= \overline{u}_i \\ \overline{v}_{2i} &= A\overline{v}_{1i} \\ \overline{v}_{3i} &= B\overline{v}_{2i}, \end{aligned}$$

and each pair of corresponding blocks belonging to different encodings forms a linear half-rate invertible code of type $(2k, k, d)$. These codes are described by matrices A, B, and the product matrix $C = BA$ relating $\overline{v}_{1i}$ and $\overline{v}_{3i}$ ($\overline{v}_{3i} = C\overline{v}_{1i}$). The minimum distances of these codes will be called d_{12}, d_{23} and d_{13}, where the subscripts indicate the encodings to which the two blocks forming the codeword belong.

Indicating with "$\diamond$" the block concatenation operator, we have

$$\overline{v}_{2i} \diamond \overline{v}_{3i} = \begin{pmatrix} A \\ C \end{pmatrix} \overline{v}_{1i} = R\overline{v}_{1i} \tag{1}$$

where

$$R \stackrel{\text{def}}{=} \begin{pmatrix} A \\ C \end{pmatrix}$$

is a $2k \times k$ block matrix. From (1) it follows that $\overline{v}_{1i} \diamond \overline{v}_{2i} \diamond \overline{v}_{3i}$ is a codeword of a $(3k, k, d_{123})$ linear code described by matrix R. We shall indicate with d_{123} its minimum Hamming distance. By extending the usual notation (n, k, d) used for linear codes, we shall denote a family of codes having these minimum distances with $[3k,\ k,\ (d_{12},\ d_{23},\ d_{13}),\ d_{123}]$.

For reasons that will become apparent during the description of the decoding procedures, we say that a family of three half-rate linear invertible codes of type $[3k,\ k,\ (d_{12},\ d_{23},\ d_{13}),\ d_{123}]$ is *optimum* if no other family of type $[3k,\ k,\ (d'_{12},\ d'_{23},\ d'_{13}),\ d'_{123}]$ exists such that:

$$\begin{gathered} [(d_{12} < d'_{12}) \wedge (d_{23} \leq d'_{23}) \wedge (d_{13} \leq d'_{13})] \vee \\ [(d_{12} \leq d'_{12}) \wedge (d_{23} < d'_{23}) \wedge (d_{13} \leq d'_{13})] \vee \\ [(d_{12} \leq d'_{12}) \wedge (d_{23} \leq d'_{23}) \wedge (d_{13} < d'_{13})] \vee \\ [(d_{12} = d'_{12}) \wedge (d_{23} = d'_{23}) \wedge (d_{13} = d'_{13}) \wedge (d_{123} < d'_{123})] \end{gathered} \tag{2}$$

where the notation $\wedge$ and $\vee$ denote logical "and" and "or", respectively.

This relation gives the precedence to the distances d_{12}, d_{23} and d_{13} over the global distance d_{123}. In fact, as we shall see, there are effective correction procedures that rely only on the codes formed by two of the encodings, whose performance is described by d_{12}, d_{23} and d_{13} only, and that are particularly attractive because of their simplicity and because of their ability to cope with burst errors.

A. Determination of Third Order Codes

From (1) it follows that the matrix R can be used to describe a third order code. The search for optimum R is based on the following theorems, whose proofs are again omitted.

Theorem 3: If R is the matrix describing a third order code $[3k,\ k,\ (d_{12},\ d_{23},\ d_{13}),\ d_{123}]$, then for any $0 < j < d_{123}$ the sum (modulo 2) of j columns of R contains at least $d_{123} - j$ elements equal to 1.

Theorem 4: If there exist a d such that, for any $0 < j < d$, the sum (modulo 2) of any j columns of matrix R contains at least $d - j$ elements equal to 1, then the d_{123} of the third order code described by R is at least d.

The starting point of the search for codes having the desired properties is the matrix B of a half-rate linear invertible code having minimum distance d_{23}. The matrices A and $C = BA$ are then built in parallel, column by column. When a column a_j of A is chosen, the corresponding column c_j of C is given by $c_j = Ba_j$. The search is again equivalent to a traversal of a tree whose nodes represent

TABLE II.

CHARACTERISTICS OF QUASI-OPTIMUM CODES FOR THIRD ORDER SCHEMES DETERMINED BY THE AUTHORS.

$[3k, k, (d_{12}, d_{23}, d_{13}), d_{123}]$
$[15, 5, (4, 4, 4), 7]$
$[18, 6, (4, 4, 4), 7]$
$[21, 7, (4, 4, 4), 7]$
$[24, 8, (5, 5, 5), 8]$
$[27, 9, (5, 5, 5), 9]$
$[30, 10, (6, 6, 6), 9]$
$[33, 11, (6, 6, 6), 10]$
$[36, 12, (6, 6, 6), 10]$
$[39, 13, (6, 6, 6), 11]$
$[42, 14, (6, 6, 6), 11]$
$[45, 15, (7, 7, 7), 11]$
$[48, 16, (7, 7, 7), 12]$

ordered sets of columns that have been chosen. A pair of columns a_j, c_j can be added to a set of already determined columns $a_1, \ldots, a_{j-1}$; $c_1, \ldots, c_{j-1}$ if the following conditions are satisfied:

1. a_j is linearly independent of all the preceding columns $a_1, a_2, \ldots, a_{j-1}$, so that $\det A = 1$, as required by invertibility of A;
2. Each of the sums of a_j with p preceding columns $a_1, a_2, \ldots, a_{j-1}$, where $0 \leq p < d_{12} - 1$, contains at least $d_{12} - p - 1$ elements equal to 1, as required by Theorem 1;
3. Each of the sums of c_j with p preceding columns $c_1, c_2, \ldots, c_{j-1}$, where $0 \leq p < d_{13} - 1$ contains at least $d_{13} - p - 1$ elements equal to 1, as required by Theorem 1;
4. Each of the sums of the column $a_j \diamond c_j$ with p preceding columns $a_1 \diamond c_1, a_2 \diamond c_2, \ldots, a_{j-1} \diamond c_{j-1}$, with $0 \leq p < d_{123} - 1$, contains at least $d_{123} - p - 1$ elements equal to 1, as required by Theorem 3.

Every node has as many children as there are valid columns that can be added to the partially determined matrices A and C associated with the node. The solutions are represented by leaves having depth k, meaning that all the columns have been successfully determined.

The time required by this method to complete the search for the matrices grows exponentially with the dimension k of the matrices, in analogy with the behavior of the previously described method for second order codes.

Tables II and III show some of the third order codes that have been found. For these codes, it has not been possible to prove the optimality in the sense of (2), and therefore they will be referred to as *quasi-optimum*.

B. The Correction Procedure

The correction procedure is an extension of the algorithm for simple second order schemes. In the case of point-to-point communications, the transmitter *TX* and the receiver *RX* apply the following procedure:

1. On receipt of the first encoding of the channel message, *RX* verifies the correctness of the received message on the basis of the FCS. If no error is detected in the received message, it is accepted, and no further retransmission is required. Otherwise, *XR* issues a NAK message asking for a new encoding of the channel message.
2. *TX* sends the second encoding of the channel message. *RX* tries first to reconstruct the original message from this second encoding only using the invertibility of matrix A. The reconstructed message is checked using its FCS and, if no error is detected, it is accepted. Otherwise, *RX* tries to correct the first and second encodings jointly on the basis of the code $(2k, k, d_{12})$ described by matrix A. If the FCS subsection of the message resulting from the joint correction shows no detectable error, the corrected message is accepted, otherwise *RX* requests the transmission of the third encoding by issuing a NAK message.
3. *TX* sends the third encoding of the channel message. *RX* first reconstructs the original message from this third encoding only, using the invertibility of matrix C. If the reconstructed message is found to be free from errors, it is accepted. Otherwise, *RX* tries to correct the three received encodings jointly, by using one of the decoding procedures described below. If the FCS of the message produced by the correction procedure shows no detectable error, the message is accepted; otherwise, *RX* asks for a new encoding of the message to be sent.
4. From this moment on, *TX* transmits cyclically the successive encoding of the channel message. *RX* tries first to reconstruct the message from the last received encoding. If the FCS shows no detectable errors, the message is accepted. Otherwise, *RX* performs the joint correction of the last three received encodings, ignoring the preceding messages. If the FCS shows no detectable errors, the message is accepted, otherwise *RX* requests another encoding of the message, and step 4 is iterated until *RX* detects no errors.

Better results could be obtained by using the information content of all the received messages, but this would require buffer sizes larger than three messages and more complex correction algorithms.

In the case of continuous broadcast, the described procedure can still be used, the only exception being that *TX* transmits the three encodings cyclically and *RX* can be connected after the second or third transmission. Notice also that the correction algorithm in the case of broadcast are one important reason why the third order codes introduced in this paper are inherently different from the other GH-ARQ codes studied so far. Indeed, any pair of different encodings forms a $(2k, k, d)$ code, and the receiver can attempt the correction as soon as it has received two differently encoded messages. For example, if the receiver begins to listen when the second encoding is transmitted and the received message contains errors, the successive received encoding will be the third one, and the correction will be attempted on the basis of the code $(2k, k, d_{23})$ described by matrix B, without having to wait for the first encoding to be received.

TABLE III.

MATRICES OF SOME QUASI-OPTIMUM THIRD ORDER CODES, GIVEN LISTING THE COLUMNS IN HEXADECIMAL NOTATION.

code		matrices
$[27, 9, (5, 5, 5), 9]$	A :	017 02D 04B 072 08E 0D8 12E 153 1BF
	B :	00F 033 055 06A 096 0AB 0DB 0ED 117
	C :	0FF 09B 08D 0D5 0E1 0CA 1B0 166 1C4
$[39, 13, (6, 6, 6), 11]$	A :	001F 006E 00E5 01B8 022B 043C 0686 0833 0A9E 0DFC 1047 1596 1E9B
	B :	001F 0067 00AB 00D5 012D 0156 01B7 01DB 01EE 022E 0433 0836 1039
	C :	012B 00F8 018E 009B 03D5 0436 070A 0835 0AF7 0D82 115D 15DE 1E49
$[45, 15, (7, 7, 7), 11]$	A :	005F 01AE 06A3 08B5 0967 0C72 10ED 1237 20F6 2353 2F1F 34FB 429D 4DBB 7F0A
	B :	003F 01C7 02D9 036A 03B4 04EB 0571 059D 067C 0727 07D2 08F2 304B 5057 702E
	C :	068E 077E 007B 08D6 0C35 0CC9 35C0 3312 54FA 5631 5DD3 62EF 70AC 7922 1CE4

The joint correction of the three encodings by receiver RX may be performed at different levels of complexity and correction capability. The purpose of any correction algorithm is to give as output, for each set of received blocks $\tilde{v}_{1i}$, $\tilde{v}_{2i}$, $\tilde{v}_{3i}$, the block $\hat{u}_i$ that represents the best estimate of the original source block $\overline{u}_i$. The reconstructed message is formed by the concatenation of the blocks $\hat{u}_i$, $1 \leq i \leq l_1 + l_2$, and can then be checked using its FCS, formed by blocks $\hat{u}_i$, $l_1{+}1 \leq i \leq l_1{+}l_2$. In the following, we shall examine several correction algorithms and highlight their advantages and disadvantages.

C. Standard Minimum Distance Decoding

Let $\hat{v}_{1i}$, $\hat{v}_{2i}$, $\hat{v}_{3i}$ be the three blocks that represent the three different channel encodings of a candidate decoded block $\hat{u}_i$. With the assumption of a binary, memoryless, symmetric channel, the optimum $\hat{u}_i$ minimizes the Hamming distance

$$\begin{aligned}\mathrm{dist}(\tilde{v}_{1i} \diamond \tilde{v}_{2i} \diamond \tilde{v}_{3i}, \hat{v}_{1i} \diamond \hat{v}_{2i} \diamond \hat{v}_{3i}) = \\ \mathrm{dist}(\tilde{v}_{1i}, \hat{v}_{1i}) + \mathrm{dist}(\tilde{v}_{2i}, \hat{v}_{2i}) + \mathrm{dist}(\tilde{v}_{3i}, \hat{v}_{3i}).\end{aligned}$$

The decoding strategy that choses such an $\hat{u}_i$ is known as minimum distance decoding, or MD for short. If the channel is not memoryless, other correction methods can perform better than MD even for small values of burst error probability, while performing comparably on memoryless channels.

D. Majority of Conjectures

For any triplet of corresponding received blocks $\tilde{v}_{1i}$, $\tilde{v}_{2i}$, $\tilde{v}_{3i}$ belonging to the tree encodings, seven conjectures can be formulated.

1. $\tilde{v}_{1i}$ is correct;
2. $\tilde{v}_{2i}$ is correct;
3. $\tilde{v}_{3i}$ is correct;
4. the block obtained from $\tilde{v}_{1i}$ and $\tilde{v}_{2i}$ applying the minimum distance correction of the code $(2k, k, d_{12})$ described by matrix A is correct;
5. the block resulting from the similar joint correction of $\tilde{v}_{2i}$ and $\tilde{v}_{3i}$ is correct;
6. the block resulting from the similar joint correction of $\tilde{v}_{3i}$ and $\tilde{v}_{1i}$ is correct;
7. the block resulting from the minimum distance joint correction of $\tilde{v}_{1i}$, $\tilde{v}_{2i}$ and $\tilde{v}_{3i}$ by using the $(3k, k, d_{123})$ code is correct.

The *majority of conjectures* decoding technique, or MC, selects as the corrected block $\hat{u}_i$ the block indicated by the relative majority of the seven conjectures, beaking ties arbitrarily. Thus, for example, if all the conjectures are different with the exception of two of them that coincide, the indication of these will be used as the corrected block. A conjecture is said to be wrong if it indicates a block different from the original one $\overline{u}_i$. Since the probability that two or more wrong conjectures coincide indicating the same candidate block is very small, it is very unlikely that two candidate blocks receive the same number of votes. The strategy used to choose between two equally voted candidate blocks is therefore irrelevant from the performance point of view. Remember that after all the blocks of the message have been reconstructed in this way, the correctness of the result is checked using the FCS included at the end of the message.

E. Majority of Six Conjectures

The determination of conjecture 7, based on the minimum distance correction of the code $(3k, k, d_{123})$ having a syndrome table of size 2^{2k}, is computationally expensive. This consideration suggests the use of the first six conjectures only, ignoring the seventh one. The selection of the conjecture to be used for the correction will be made also in this case with the relative majority method. This decoding technique will be called the M6C decoding, from Majority of Six Conjectures. M6C decoding is computationally simpler than MD decoding, as it only needs three syndrome tables of size 2^k.

This correction method, although inferior to MD decoding for binary symmetric memoryless channels, performs better for channels affected by burst errors. In fact, when a block of one of the three encodings is affected by a long error burst, MD decoding is generally unable to reconstruct the original message, due to the large total number of errors. On the other hand, M6C decoding is often able to reconstruct the original block on the basis of the three conjectures derived from the two blocks belonging to the uncorrupted encodings and from their joint correction.

F. Comparison Between MD and M6C Decoding

It is possible to derive analytical formulas giving the correction capability of both MD and M6C decodings. We shall indicate with $P_e(i_1, i_2, i_3)$ the probability that the three corresponding received blocks of the three encodings contain i_1, i_2, i_3 errors, respectively. In the case of a binary symmetric memoryless channel, this quantity is given by:

$$P_e(i_1, i_2, i_3) = \binom{k}{i_1}\binom{k}{i_2}\binom{k}{i_3} p^{i_1+i_2+i_3}(1-p)^{3k-i_1-i_2-i_3},$$

where p is the bit error probability of the channel. Let

$$t_3 \stackrel{\text{def}}{=} \left\lfloor \frac{d_{123}-1}{2} \right\rfloor$$

be the maximum number of errors that the MD decoding of the $(3k,\ k,\ d_{123})$ code can correct, and let c_{MD} be the function whose value is 1 if the error pattern is correctable with MD decoding, and 0 otherwise:

$$c_{\text{MD}}(i_1, i_2, i_3) \stackrel{\text{def}}{=} \begin{cases} 1 \text{ if } i_1 + i_2 + i_3 \leq t_3 \\ 0 \text{ otherwise} \end{cases}$$

The probability that MD decoding successfully reconstructs the original information from three corresponding received blocks is therefore given by

$$P_{\text{MD}}\{correct\} = \sum_{i_1=0}^{k}\sum_{i_2=0}^{k}\sum_{i_3=0}^{k} P_e(i_1, i_2, i_3) c_{\text{MD}}(i_1, i_2, i_3). \tag{3}$$

For M6C decoding, we define a function $c_{\text{M6C}}(i_1, i_2, i_3)$ similar to the one defined for MD decoding. Since it is very unlikely that two different conjectures leading to an erroneous correction coincide, we can safely assume that the correction is possible whenever there are at least two coinciding conjectures leading to the correct result. If $d_{12} = d_{13} = d_{23}$, as is the case for the codes presented in this paper, let

$$t_2 \stackrel{\text{def}}{=} \left\lfloor \frac{d_{12}-1}{2} \right\rfloor$$

be the maximum number of errors that can be corrected by one of the $(2k, k, d)$ codes formed by a pair of corresponding blocks. Let

$$r(i_1, i_2, i_3) \stackrel{\text{def}}{=} \sum_{j=1}^{3} \delta_{i_j=0} + \sum_{j=2}^{3}\sum_{l=1}^{j-1} \delta_{i_j+i_l \leq t_2}$$

be the number of correct conjectures, that is, conjectures that correctly guess the original value of the block. The symbol δ_R denotes 1 if relation R is true and 0 otherwise. We define $c_{\text{M6C}}(i_1, i_2, i_3)$ as

$$c_{\text{M6C}}(i_1, i_2, i_3) \stackrel{\text{def}}{=} \begin{cases} 1 \text{ if } r(i_1, i_2, i_3) \geq 2 \\ 0 \text{ otherwise} \end{cases}$$

and $P_{\text{M6C}}\{correct\}$ is therefore given by

$$P_{\text{M6C}}\{correct\} = \sum_{i_1=0}^{k}\sum_{i_2=0}^{k}\sum_{i_3=0}^{k} P_e(i_1, i_2, i_3) c_{\text{M6C}}(i_1, i_2, i_3). \tag{4}$$

In order to obtain information about the behavior of the codes on channels affected by error bursts, it is necessary to model the probabilistic behavior of these channels and compute $P_e(i_1, i_2, i_3)$.

Since message blocks are the basic unit of correction, every block being corrected independently, it is not necessary to consider error bursts longer than one block. In the following we shall suppose that the average length of an error burst is much greater than the length of a block. This is indeed the case in the majority of applications as, for example, communications over telephone lines. Consequently, we shall assume for simplicity that a block is either completely affected by a burst error, or is totally exempt from the effects of bursts, and contains only uncorrelated errors.

Therefore, we define the *burst error probability* q as the probability that a block is affected by a burst error. We will assume that the error probability of a bit belonging to a block not corrupted by burst errors is p, while the error probability of a bit belonging to a block corrupted by a burst error is $1/2$, which represents the total loss of correlation between the sent and received information. Under these assumptions, $P_e(i_1, i_2, i_3)$ is given by:

$$P_e(i_1, i_2, i_3) = \prod_{j=1}^{3} \binom{k}{i_j} \left[(1-q)p^{i_j}(1-p)^{k-i_j} + q2^{-k}\right].$$

Substituting this equation in (3) and (4), it is possible to evaluate the different behavior of the two decoding schemes in the presence of error bursts .

V. The Synchronization Problem

If efficient data communication is to be achieved over a communication channel, some means of keeping the receiver synchronized with the transmitter should be introduced. In some types of physical channels, the synchronization is enforced using the bit stream transmitted over the channel. This is the case, for example, in communication systems using FSK (Frequency Shift Keying) encoding to transmit data over telephone lines. In this case, an error corrupting one bit of the message can often cause the loss of the bit or message synchronization, and it is therefore necessary to introduce some error-free synchronization mechanism if the advantages of the codes presented in this paper are to be fully exploited. Precisely, the method chosen for encoding the messages should satisfy two important requirements:

1. a one-bit error in the channel should result in a one-bit error in the received message;
2. the probability that a message is received with an incorrect length should be as small as possible, since this would prevent the decoding algorithm from working effectively.

In the HDLC and BSC transmission protocols, a special bit sequence or a special character are respectively used to mark the end of a message, and bit or character stuffing mechanisms are used to ensure that these marker sequences do not appear inside a message. However, the bit or character stuffing algorithms violate the first of the preceding conditions, as an error occurring in a sequence modified by the stuffing can lead the receiver to perform the unstuffing incorrectly. This may cause the loss of synchronism with the transmitter's operations, resulting in a very high number of differences between the original and the received message.

To delimit the message properly it is therefore necessary to use a delimiter block $\overline{v}_d$ different from all the blocks used for information transmission, thus making the stuffing mechanism unnecessary. To satisfy the second condition, moreover, the block $\overline{v}_d$ should have a high Hamming distance from all the blocks used for information encoding. This serves the purpose of reducing the probability that a message block is mistakenly interpreted as the delimiter block or vice versa, in turn reducing the probability that the message is received with an incorrect length.

The frame composed by the source message and its FCS is therefore divided into a sequence of m bits long source blocks $\overline{u}_i$ (see Fig. 1). Each source block $\overline{u}_i$ is then encoded using the *pre-coding function* φ into a k bit channel block $\overline{v}_i = \varphi(\overline{u}_i)$. The sequence of the $\overline{v}_i$ blocks forms the first encoding. In order to assign a different channel block to each of the 2^m source blocks, and to have one distinguished block $\overline{v}_d$ that serves only as a message delimiter, k should be larger than m. To limit the amount of redundancy introduced by the pre-coding function, we will choose $k = m + 1$.

The function φ should be such that all the blocks $\overline{v}$ used in any of the encodings have the following *channel* properties:

1. The Hamming distance between $\overline{v}$ and the packet delimiter $\overline{v}_d$ should be greater than or equal to a certain value d_d.
2. $\overline{v}$ should contain at least g internal transitions $0 \to 1$ or $1 \to 0$. This enables the transmitter and the receiver to remain in bit synchronism using the transitions of the bit stream. This is important when the modulation employed does not have internal clocking capabilities, as in FSK.

The procedure to construct the function φ for given d_d and g can be outlined as follows:

1. A table of channel blocks to be used for information encoding is constructed. The table has two or three columns, for second and third order schemes respectively, and 2^k rows. The first column contains all the k-bit blocks listed in increasing order if considered as k-bit binary numbers, from 0 to $2^k - 1$.
   If $\overline{v}_{ij}$ is the block in column i and row j, the second column is obtained from $\overline{v}_{2j} = A\overline{v}_{1j}$, and, in the case of third order schemes, the third column is obtained from $\overline{v}_{3j} = B\overline{v}_{2j}$.
2. If all the blocks in a row satisfy the two channel properties above mentioned, then the row is labelled "good;" otherwise, it is labelled as "bad."
3. If the final number of good rows is less than 2^m, the values of d_d and g place too strict restrictions on the function φ. This restrictions can be solved by decreasing their values, or by using a smaller value for m, and returning to step 2.
4. Otherwise, a function mapping the set of m-bit source blocks into the set of k-bit blocks appearing in the first column of the good rows can be used as a pre-coding function φ.

A. Some Codes

Table IV presents some of the most interesting codes found, together with the value of d_d and g. A sequence of k bits equal to 1 was chosen as the delimiter block. This sequence has the property that it is invariant under circular shift of its bits, and therefore if two such consecutive blocks are used to delimit a message, the receiver is able to detect the end-of-message even when the bit synchronism has been lost. The pre-coding functions are not listed, for the sake of brevity, but can be obtained as previously outlined.

B. The Reconstruction of the Original Message

To reconstruct the source message, the sequence of blocks $\overline{v}_{1i}$ forming the first encoding is first obtained using the decoding algorithms outlined in the preceding sections. Then, if $\overline{v}_{1i}$ is an element of the image of φ, the source block is obtained through $\overline{u}_i = \varphi^{-1}(\overline{v}_{1i})$; otherwise an error is detected and the receiver requests another encoding of the channel message, using the retransmission procedure illustrated in the preceding sections.

VI. Throughput of the Cyclic GH-ARQ Schemes

We have used computer simulation to evaluate the throughput of the cyclic GH-ARQ schemes and to compare it to the throughput of other schemes previously studied, e.g., by Krishna and Morgera [22]. The simulation models is based on the following simplifying assumptions:

1. The FCS used has a perfect error detection capability: it never fails to detect errors present in the message.
2. It is always possible to correctly decode the header of the messages.
3. The message is always received with the correct length.
4. In the computation of throughput, the lengths of the header and of the FCS are not accounted for neither in

TABLE IV.
SOME SECOND AND THIRD ORDER CODES WITH PRE-CODING FUNCTION, GIVEN LISTING THE COLUMNS OF THE MATRICES IN HEXADECIMAL NOTATION.

code	m	g	d_d		matrices
$(16, 8, 5)$	7	3	3		0F 33 55 6A 96 AE ED F1
$(22, 11, 7)$	10	4	4		03F 1C7 2D9 36A 3B4 4EB 571 59D 67C 727 7D2
$[27, 9, (5, 5, 5), 9]$	8	3	3	A :	017 02D 04B 072 08E 0D8 12E 153 1BF
				B :	00F 033 055 06A 096 0AB 0DB 0ED 117
				C :	0FF 09B 08D 0D5 0E1 0CA 1B0 166 1C4
$[39, 13, (6, 6, 6), 11]$	12	4	5	A :	001F 006E 00E5 01B8 022B 043C 0686 0833 0A9E 0DFC 1047 1A12 1F06
				B :	001F 0067 00AB 00D5 012D 0156 01B7 01DB 01EE 022E 0433 0836 1039
				C :	012B 00F8 018E 009B 03D5 0436 070A 0835 0AF7 0D82 115D 1B6B 1F30

the simple ARQ transmission scheme nor in the GH-ARQ scheme, nor in scheme [22].

5. To make the comparisons homogeneous, we study a case where the pre-coding function φ is not necessary and is taken to be the identity function, as in the simple second and third order schemes.

Assumption 1 is reasonable, when the FCS is long enough, so that the probability of undetected errors can be capt to very small values. Assumption 2 derives from the fact that the header is protected by a powerful code, and thus the probability of its incorrect decoding is very small. Assumption 3 is valid when the synchronization is achieved with some reliable mechanism.

As a measure of the performance of the code and transmission scheme we take the throughput, given by the average value of the ratio

$$\eta \stackrel{\text{def}}{=} \frac{N_s}{N_c},$$

where N_s is the length in bits of the message to be transmitted, and N_c is the average number of bits that must be transmitted through the channel before the decoder is able to reconstruct the message. If no pre-coding function is used, this ratio equals the reciprocal of the average number of times ζ that a message is transmitted through the channel, including the initial transmission.

A. Results

Tables V, VI and VII show the values of $\zeta = 1/\eta$ obtained for the simple ARQ transmission scheme, for the schemes proposed in [22] and for several cyclic GH-ARQ transmission schemes presented in this paper. For the third order codes, the M6C decoding technique was used. The code A2 proposed by Krishna and Morgera has not been included in the comparisons, since it is basically a less powerful version of code A3, in which the fourth encoding is not used. All the figures have been obtained by counting the number of copies that had to be sent to be able to correctly decode 200 messages at the receiving end of the channel. When the number of required retransmissions was greater than ten the simulation stopped, as the performance was too poor for its exact value to be of any interest. Fig. 2, 3 and 4 display in graphical form the comparison between the best code proposed by Krishna and Morgera and the best third order code presented in this paper, again using M6C decoding. The comparison was limited to one code per family for the sake of clarity.

Fig. 5 shows a comparison between MD and M6C decoding techniques for a channel affected by error bursts. The results of the comparison for a channel not affected by error bursts are not shown, since for this code the difference in behavior between the two techniques is not noticeable for meaningful values of the bit error rate, although the MD decoding is in principle superior.

VII. CONCLUSIONS

The results of the comparison between the various codes show that the simple ARQ retransmission scheme is remarkably inferior to both the schemes presented by Krishna and Morgera and the schemes presented in this paper. Moreover, with the parameters chosen for the simulation, for values of the error rate $p < 10^{-4}$, $q < 10^{-4}$, the codes proposed by Krishna and Morgera and those presented by the authors have very similar performances, as very seldom the receiver fails to reconstruct the original message after the second encoding has been received.

When the bit and burst error rates grow above that threshold, the differences between the various schemes become noticeable. In absence of burst errors ($q = 0$), Table VI and Fig. 2 show that the schemes proposed in this paper perform generally better until the number of average retransmission is so high that both families of schemes become inadequate. For example, Fig. 2 shows that the KM-3

TABLE V.

ABBREVIATIONS USED FOR SOME CODES WHOSE THROUGHPUT HAS BEEN ESTIMATED. k IS THE BLOCK LENGTH, t_2, t_3 AND t_4 ARE THE NUMBER OF ERRORS THAT CAN BE CORRECTED WHEN RESPECTIVELY 2, 3 OR 4 DIFFERENT ENCODINGS OF THE BLOCK ARE AVAILABLE, AND KM STANDS FOR THE CODES PROPOSED BY KRISHNA AND MORGERA IN [22].

abbrev.	type	code	k	t_2	t_3	t_4
KM-1	KM (A1)	$(15, 5, 5)$	5	1	2	—
KM-3	KM (A3)	$(24, 6, 9)$	6	1	2	4
KM-4	KM (A4)	$(28, 7, 10)$	7	1	2	4
KM-5	KM (A5)	$(32, 8, 9)$	8	1	2	4
s8	second order	$(16, 8, 5)$	8	2	—	—
s11	second order	$(22, 11, 7)$	11	3	—	—
t9	third order	$[27, 9, (5, 5, 5), 9]$	9	2	4	—
t13	third order	$[39, 13, (6, 6, 6), 11]$	13	2	5	—
t15	third order	$[45, 15, (7, 7, 7), 11]$	15	3	5	—

TABLE VI.

AVERAGE NUMBER OF TRANSMISSIONS ζ OF THE CODES AS A FUNCTION OF THE ERROR PROBABILITY p, FOR MESSAGES 512 BYTES LONG AND BURST ERROR PROBABILITY $q = 0$.

($q = 0$)		KM				second order		third order		
p	ARQ	KM-1	KM-3	KM-4	KM-5	s8	s11	t9	t13	t15
0.00001	1.04	1.02	1.03	1.03	1.02	1.02	1.03	1.03	1.03	1.03
0.00003	1.13	1.11	1.10	1.11	1.10	1.10	1.11	1.10	1.11	1.11
0.0001	1.50	1.31	1.32	1.32	1.32	1.32	1.33	1.34	1.35	1.33
0.0003	3.42	1.69	1.73	1.73	1.75	1.74	1.71	1.75	1.74	1.74
0.001	> 10	2.01	2.04	2.03	2.03	1.97	1.97	1.98	1.97	1.98
0.003	> 10	2.27	2.35	2.43	2.40	2.00	2.00	2.00	2.04	2.00
0.01	> 10	3.52	3.29	3.53	3.57	2.38	2.03	2.26	2.60	2.09
0.02	> 10	> 10	4.05	4.19	4.32	7.98	2.42	3.45	5.91	2.72
0.03	> 10	> 10	4.69	5.56	7.72	> 10	5.34	9.98	> 10	5.15

TABLE VII.

AVERAGE NUMBER OF TRANSMISSIONS ζ OF THE CODES FOR MESSAGES 512 BYTES LONG AND ERROR PROBABILITIES $p = q$.

	KM				second order		third order		
p, q	KM-1	KM-3	KM-4	KM-5	s8	s11	t9	t13	t15
0.00001	1.03	1.03	1.03	1.03	1.03	1.03	1.03	1.03	1.03
0.00003	1.14	1.13	1.11	1.11	1.11	1.11	1.10	1.11	1.11
0.0001	1.50	1.42	1.40	1.41	1.39	1.38	1.36	1.37	1.34
0.0003	2.29	2.23	2.11	2.17	2.01	1.89	1.94	1.89	1.89
0.001	6.17	4.07	4.42	5.10	4.02	3.13	2.56	2.42	2.35
0.003	> 10	7.69	> 10	> 10	> 10	9.22	2.96	2.84	2.84
0.01	> 10	> 10	> 10	> 10	> 10	> 10	3.26	3.36	3.29
0.02	> 10	> 10	> 10	> 10	> 10	> 10	7.73	> 10	7.15

and t15 codes have similar throughput for $p < 10^{-3}$. For higher values of the bit error rate, code t15 is superior, until p reaches the value of approximately 0.025. Beyond that value, the code CM-A3 takes the lead, but the throughput is so low (less that 0.2) that this has scarce practical importance: using shorter packets would very likely be a better idea than selecting one code instead of the other. Note also that in the absence of burst errors the code s11 provides a simpler but equally effective alternative to the third order codes.

In the presence of burst errors, the superiority of the third order codes presented in this paper becomes even greater, as Table VII and Fig. 3 and 4 clearly show. The threshold effect previously mentioned disappears, and the third order codes have a higher throughput for all the range of error rates that has been studied. The reason of the increased difference in throughput lies in the use of the M6C decoding, which often enables the reconstruction of the message if the three encodings have not been corrupted by bursts in the same positions. On the other hand the second order schemes, although superior to the KM codes in the useful range of error rates, do not show a similar relative increase in performance, as they use the classical minimum distance decoding.

Fig. 5 shows the importance of M6C decoding in the presence of burst errors. As M6C is also computationally simpler than MD decoding, as previously remarked, it offers a very interesting alternative for the decoding of the messages.

Concluding, the transmission schemes and codes presented in this paper proved to couple an attractive performance with simple encoding and decoding techniques which are particularly apt for channels affected by error bursts.

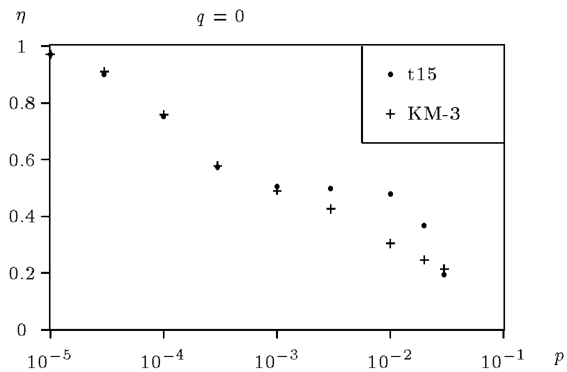

Fig. 2. Throughput η of the codes KM-3 and t15 versus the bit error probability p, for messages 512 bytes long and burst error probability $q = 0$.

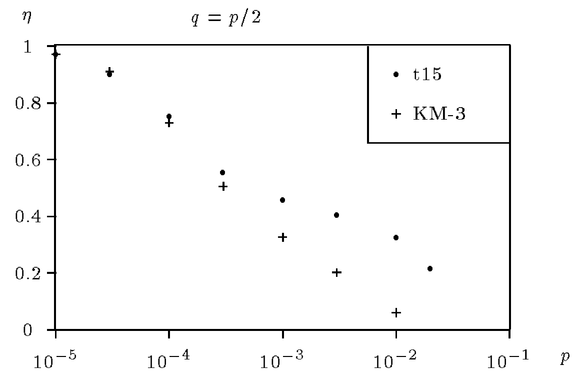

Fig. 3. Throughput η of the codes KM-3 and t15 for messages 512 bytes long and burst error probability q equal to $p/2$.

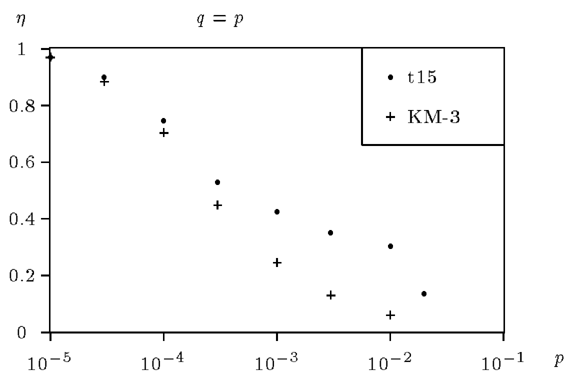

Fig. 4. Throughput η of the codes KM-3 and t15 for messages 512 bytes long and burst error probability q equal to the bit error probability p.

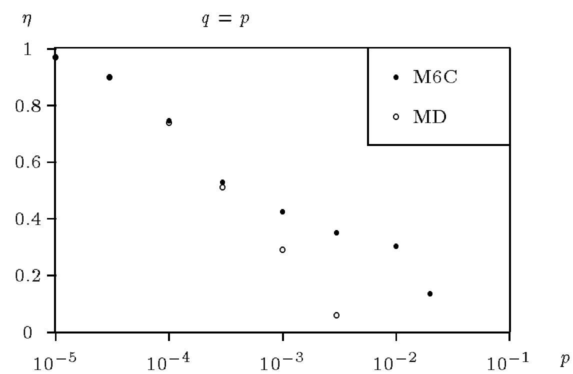

Fig. 5. Throughput η of code t15: comparison between MD and M6C decoding for burst error probability q equal to the bit error probability p, and messages 512 bytes long.

References

[1] D. M. Mandelbaum, "Adaptative-feedback coding scheme using incremental redundancy," *IEEE Trans. Inform. Theory*, vol. IT-20, pp. 388-389, May 1974.

[2] F. J. MacWilliams and N. J. A. Sloane, "The theory of error correcting codes," *North-Holland*, 1977.

[3] J. J. Metzner, "Improvements in block-retransmission schemes," *IEEE Int. Symp. Inform. Theory*, Ithaca, NY, Oct. 1977.

[4] J. J. Metzner, "Improvements in block-retransmission schemes," *IEEE Trans. on Comm.*, vol. COM-27, pp. 525-532, 1979.

[5] T. C. Ancheta, "Convolutional parity check automatic repeat request," *IEEE Int. Symp. Inform. Theory*, Grignano, Italy, 1979.

[6] S. Lin and J. S. Ma, "A hybrid ARQ system with parity retransmission for error correction," *IBM Res. Rep. 7478 (32232)*, Jan. 1979.

[7] S. Lin and S. Yu, "SPREC - An effective hybrid-ARQ scheme," *IBM Res. Rep. 7591 (32852)*, Apr. 1979.

[8] S. Lin and D. J. Costello, Jr., "Error Control Coding: Fundamentals and Applications," *Prentice Hall*, Englewood Cliffs, NJ, 1983.

[9] S. Lin, D.J. Costello and M. J. Miller, "Automatic repeat-request error control schemes," IEEE Commun. Mag., vol 12, pp. 5-17, Dec. 1984.

[10] D. Chase, P. D. Mullers, and J. K. Wolf, "Application of code combining to a selective-repeat ARQ link," in *Proc. MILCOM '85, Conf. Rec.*, vol. 1, 1985, pp. 247-252.

[11] D. Chase, "Code combining: A maximum-likelihood decoding approach for combining an arbitrary number of noisy packets," *IEEE Trans. Commun.*, vol. COM-33, pp. 385-393, May 1985.

[12] C. Lau and C. Leung, "Performance analysis of a memory ARQ scheme with soft decision detectors," *IEEE Trans. Commun.*, vol. COM-34, pp. 827-832, Aug. 1986.

[13] P. S. Sindhu, "Retransmission error control with memory," *IEEE Trans. Commun.*, vol. COM-25, pp. 473-479, May 1977.

[14] S. Kallel and D. Hacoun, "Sequential decoding with ARQ and code combining: a robust hybrid FEC/ARQ System," *IEEE Trans. Commun.*, vol. COM-36, pp.773-779, July 1988.

[15] N. Shacham, "Performance of ARQ with sequential decoding over one-hop and two-hop radio links," *IEEE Trans. Commun.*, vol. COM-31, pp. 1172-1180, Oct.1983.

[16] N. Shacham, "ARQ with sequential decoding of packetized data: Queuing analysis," *IEEE Trans. Commun.*, vol. COM-32, pp. 1118-1127, Oct. 84.

[17] A. Drukarev and D.J. Costello, Jr., "Hybrid ARQ error control using sequential decoding," *IEEE Trans. Inform. Theory*, vol. IT-29, pp. 521-535, July 1983.

[18] P. Y. Pau and D. Haccoun, "An analysis of sequential decoding with retransmission procedures," Ecole Polytechnique de Montréal, Tech. Rep. EMP/RT-85-19, Montréal, P.Q., Canada, May 1985.

[19] P.S. Yu and S. Lin, "An efficient selective-repeat ARQ scheme for satellite channels and its throughput analysis, *IEEE Trans. Commun.*, vol. COM-29, pp. 353-363, Mar. 81.

[20] S. Lin and P.S. Yu, "A hybrid ARQ scheme with parity retransmission for error control of satellite channels," *IEEE Trans. Commun.*, vol. COM-30, pp. 1701-1719, 1982.

[21] Y. M. Wang and S. Lin, "A modified selective-repeat Type-II hybrid ARQ system and its performance analysis," *IEEE Trans. Commun.*, vol. COM-31, pp.593-607, May 1983.

[22] H. Krishna and S. D. Morgera, "A new error control scheme for hybrid ARQ systems," *IEEE Trans. Commun.*, vol. COM-35,

pp. 981-990, Oct. 1987.

[23] S. D. Morgera and H. Krishna, "Digital signal processing – Applications to Communications and Algebraic Coding Theories," Boston, Academic Press, 1989.

[24] J. Hagenauer, "Rate-compatible punctured convolutional codes (RCPC codes) and their applications," *IEEE Trans. Commun.*, vol. COM-36, pp. 389-400, Apr. 1988.

[25] S.D. Sandberg and M.B. Pursley, "Retransmission schemes for meteor-burst communications," Proceedings of the Conference on Computers and Communications, pp.246-253, March 1990.

Luca de Alfaro graduated in electrical engineering at Politecnico di Torino in 1990, where in the same year he started a Doctorate in computer science. After working for some time on error correcting codes, in 1991 he started a Ph.D. in computer science at Stanford University, where he currently is a degree candidate. His research interests include temporal and modal logic, and the formal verification of real-time systems and programs.

Angelo Raffaele Meo graduated in electrical engineering in 1958. Since 1970 he is a full professor of computer science at Politecnico di Torino. From 1979 to 1985 he was the director of *Progetto finalizzato informatica*, a national research program of CNR for the advancement of computer science involving over 2000 researchers. He is currently head of CENS (the center for digital signal processing of CNR) and CSP (*Consorzio Supercalcolo Piemonte*). He is the author of over one hundred articles on switching theory, logical design, speech synthesis and recognition.