Chapter 17
Combined Error Detection and Error-Correction

17.1 Analysis of Undetected Error Probability

Let the space of vectors over a field with q elements $\mathbb{F}_q$ of length n be denoted by $\mathbb{F}_q^n$. Let $[n, k, d]_q$ denote a linear code over $\mathbb{F}_q$ of length n symbols, dimension k symbols and minimum Hamming distance d. We know that a code with minimum Hamming distance d can correct $t = \lfloor (d-1)/2 \rfloor$ errors. It is possible for an $[n, k, d = 2t+1]_q$ linear code, which has q^{n-k} syndromes, to use a subset of these syndromes to correct $\tau < t$ errors and then to use the remaining syndromes for error detection. For convenience, let $\mathscr{C}$ denote an $[n, k, d]_q$ linear code with cardinality $|\mathscr{C}|$, and let a codeword of $\mathscr{C}$ be denoted by $\boldsymbol{c}_l = (c_{l,0}, c_{l,1}, \ldots, c_{l,n-1})$, where $0 \leq l < |\mathscr{C}|$.

Consider a codeword $\boldsymbol{c}_i$, for some integer i, which is transmitted over a q-ary symmetric channel with symbol transition probability $p/(q-1)$. At the receiver, a length n vector $\boldsymbol{y}$ is received. This vector $\boldsymbol{y}$ is not necessarily the same as $\boldsymbol{c}_i$ and, denoting $\mathrm{d}_H(\boldsymbol{a}, \boldsymbol{b})$ as the Hamming distance between vectors $\boldsymbol{a}$ and $\boldsymbol{b}$, the following possibilities may occur assuming that nearest neighbour decoding algorithm is employed:

1. (no error) $\mathrm{d}_H(\boldsymbol{y}, \boldsymbol{c}_i) \leq \tau$ and $\boldsymbol{y}$ is decoded as $\boldsymbol{c}_i$;
2. (error) $\mathrm{d}_H(\boldsymbol{y}, \boldsymbol{c}_j) > \tau$ for $0 \leq j < |\mathscr{C}|$; and
3. (undetected error) $\mathrm{d}_H(\boldsymbol{y}, \boldsymbol{c}_j) \leq \tau$ for $j \neq i$ and $\boldsymbol{y}$ is decoded as $\boldsymbol{c}_j$

Definition 17.1 A sphere of radius t centered at a vector $\boldsymbol{v} \in \mathbb{F}_q^n$, denoted by $S_q^t(\boldsymbol{v})$, is defined as

$$S_q^t(\boldsymbol{v}) = \{\boldsymbol{w} \mid wt_H(\boldsymbol{v} - \boldsymbol{w}) \leq t \text{ for all } \boldsymbol{w} \in \mathbb{F}_q^n\}. \tag{17.1}$$

It can be seen that, in an error-detection-after-correction case, $S_q^\tau(\boldsymbol{c})$ may be drawn around all $|\mathscr{C}|$ codewords of the code $\mathscr{C}$. For any vector falling within $S_q^\tau(\boldsymbol{c})$, the decoder returns $\boldsymbol{c}$ the corresponding codeword which is the center of the sphere. It is worth noting that all these $|\mathscr{C}|$ spheres are pairwise disjoint, i.e.

M. Tomlinson et al., *Error-Correction Coding and Decoding*,
Signals and Communication Technology,
DOI 10.1007/978-3-319-51103-0_17

$$\bigcup_{\substack{0\le i,j<|\mathscr{C}|\\ i\neq j}} S_q^\tau(\boldsymbol{c}_i) \cap S_q^\tau(\boldsymbol{c}_j) = \emptyset.$$

In a pure error-detection scenario, the radius of these spheres is zero and the probability of an undetected error is minimised. When the code is used to correct a given number of errors, the radius increases and so does the probability of undetected error.

Lemma 17.1 *The number of length n vectors over $\mathbb{F}_q$ of weight j within a sphere of radius τ centered at a length n vector of weight i, denoted by $N_q^\tau(n, i, j)$, is equal to*

$$N_q^\tau(n,i,j) = \sum_{e=e_L}^{e_U} \sum_{\delta=\delta_L}^{\delta_U} \binom{i}{e}\binom{e}{\delta}\binom{n-i}{j-i+\delta}(q-1)^{j-i+\delta}(q-2)^{e-\delta} \tag{17.2}$$

where $e_L = \max(0, i-j)$, $e_U = \min(\tau, \tau+i-j)$, $\delta_L = \max(0, i-j)$ and $\delta_U = \min(e, \tau+i-j-e, n-j)$.

Proof Let $\boldsymbol{u}$ be a vector of weight i and let $\sup(\boldsymbol{u})$ and $\overline{\sup}(\boldsymbol{u})$ denote the support of $\boldsymbol{u}$, and the non-support of $\boldsymbol{u}$, respectively, that is

$$\begin{aligned}\sup(\boldsymbol{u}) &= \{i \mid u_i \neq 0, \text{ for } 0 \le i \le n-1\}\\ \overline{\sup}(\boldsymbol{u}) &= \{0, 1, \ldots, n-1\} \setminus \sup(\boldsymbol{u}).\end{aligned}$$

A vector of weight j, denoted by $\boldsymbol{v}$, may be obtained by adding a vector $\boldsymbol{w}$, which has e coordinates which are the elements of $\sup(\boldsymbol{u})$ and f coordinates which are the elements of $\overline{\sup}(\boldsymbol{u})$. In the case where $q > 2$, considering the coordinates in $\sup(\boldsymbol{u})$, it is obvious that vector $\boldsymbol{v} = \boldsymbol{u} + \boldsymbol{w}$ can have more than $i - e$ non-zeros in these coordinates. Let δ, where $0 \le \delta \le e$, denote the number of coordinates for which $v_i = 0$ among $\sup(\boldsymbol{u})$ of $\boldsymbol{v}$, i.e.

$$\delta = |\sup(\boldsymbol{u}) \setminus (\sup(\boldsymbol{u}) \cap \sup(\boldsymbol{v}))|$$

Given an integer e, there are $\binom{i}{e}$ ways to generate e coordinates for which $w_i \neq 0$ where $i \in \sup(\boldsymbol{u})$. For each way, there are $\binom{e}{e-\delta}(q-2)^{e-\delta}$ ways to generate $e-\delta$ non-zeros in the coordinates $\sup(\boldsymbol{u}) \cap \sup(\boldsymbol{w})$ such that $v_i \neq 0$. It follows that $f = j - (i-e) - (e-\delta) = j - i + \delta$ and there are $\binom{n-i}{j-i+\delta}(q-1)^{j-i+\delta}$ ways to generate f non-zero coordinates such that $v_i \neq 0$ where $i \in \overline{\sup}(\boldsymbol{u})$. Therefore, for given integers e and δ, we have

$$\binom{i}{e}\binom{e}{\delta}\binom{n-i}{j-i+\delta}(q-1)^{j-i+\delta}(q-2)^{e-\delta} \tag{17.3}$$

vectors $\boldsymbol{w}$ that produce $wt_H(\boldsymbol{v}) = j$. Note that $\binom{e}{e-\delta} = \binom{e}{\delta}$.

It is obvious that $0 \le e, f \le \tau$ and $e + f \le \tau$. In the case of $j \le i$, the integer e may not take the entire range of values from 0 to τ, it is not possible to have $e < i - j$. On the other hand, for $j \ge i$, the integer $e \ge 0$ and thus, the lower limit on the value of e is $e_L = \max(0, i - j)$. The upper limit of e, denoted by e_U, is dictated by the condition $e + f = \tau$. For $j \le i$, $e_U = \tau$ since for any value of e, δ may be adjusted such that $wt_H(\boldsymbol{v}) = j$. For the case $j \ge i$, $f \ge 0$ and for any value of e, there exists at least one vector for which $\delta = 0$, implying $e_U = \tau - f = \tau + i - j$. It follows that $e_U = \min(\tau, \tau + i - j)$.

For a given value of e, δ takes certain values in the range between 0 and e such that $wt_H(\boldsymbol{v}) = j$. The lower limit of δ is obvious $\delta_L = e_L$. The upper limit of δ for $j \ge i$ case is also obvious, $\delta_U = e$, since $f \ge 0$. For the case $j \le i$, we have $e + f = e + (j - i + \delta_U) \le \tau$, implying $\delta_U \le \tau - e + i - j$. In addition, $n - i \ge j - i + \delta_U$ and thus, we have $\delta_U = \min(e, \tau - e + i - j, n - j)$.

Corollary 17.1 *For $q = 2$, we have*

$$N_2^\tau(n, i, j) = \sum_{e=\max(0,i-j)}^{\lfloor(\tau+i-j)/2\rfloor} \binom{i}{e}\binom{n-i}{j-i+e} \tag{17.4}$$

Proof For $q = 2$, it is obvious that $\delta = e$ and $0^0 = 1$. Since $e + f \le \tau$ and $f = j - i + e$, the upper limit of e, e_L, becomes $e_L \le \lfloor(\tau + i - j)/2\rfloor$.

Theorem 17.1 *For an $[n, k, d = 2t + 1]_q$ linear code $\mathscr{C}$, the probability of undetected error after correcting at most τ errors, where $\tau \le t$, in a q-ary symmetric channel with transition probability $p/(q - 1)$, is given by*

$$P_{ue}^{(\tau)}(\mathscr{C}, p) = \sum_{i=d}^{n} A_i \sum_{j=i-\tau}^{i+\tau} N_q^\tau(n, i, j) \left(\frac{p}{q-1}\right)^j (1-p)^{n-j} \tag{17.5}$$

where A_i is the number of codewords of weight i in $\mathscr{C}$ and $N_q^\tau(n, i, j)$ is given in Lemma 17.1.

Proof An undetected error occurs if the received vector falls within a sphere of radius τ centered at any codeword $\mathscr{C}$ except the transmitted codeword. Without loss of generality, as the code is linear, the transmission of the all zeros codeword may be assumed. Consider $\boldsymbol{c}_i$ a codeword of weight $i > 0$, all vectors within $S_q^\tau(\boldsymbol{c}_i)$ have weights ranging from $i - \tau$ to $i + \tau$ with respect to the transmitted all zeros codeword. For each weight j in the range, there are $N_q^\tau(n, i, j)$ such vectors in the sphere.

Following [2], if B_j denotes the number of codewords of weight j in $\mathscr{C}^\perp$, the dual code of $\mathscr{C}$, A_j may be written as

$$A_m = \frac{1}{|\mathscr{C}^\perp|} \sum_{i=0}^{n} B_i P_q(n, m, i) \tag{17.6}$$

where

$$P_q(n,m,i)=\sum_{j=0}^{m}(-1)^j q^{m-j}\binom{n-m+j}{j}\binom{n-i}{m-j} \tag{17.7}$$

is a Krawtchouk polynomial. Using (17.6) and (17.7), the probability of undetected error after error-correction (17.5) may be rewritten in terms of the weight of the codewords in the dual code.

17.2 Incremental-Redundancy Coding System

17.2.1 Description of the System

The main area of applications is two-way digital communication systems with particular importance to wireless communication systems which feature packet digital communications using a two-way communications medium. In wireless communications, each received packet is subject to multipath effects and noise plus interference causing errors in some of the received symbols. Typically forward error-correction (FEC) is provided using convolutional codes, turbo codes, LDPC codes, or algebraic block codes and at the receiver a forward error-correction decoder is used to correct any transmission errors. Any residual errors are detected using a cyclic redundancy check (CRC) which is included in each transmitted codeword. The CRC is calculated for each codeword that is decoded from the corresponding received symbols and if the CRC is not satisfied, then the codeword is declared to be in error. If such an error is detected, the receiver requests the transmitter by means of a automatic repeat request (ARQ) either to retransmit the codeword or to transmit additional redundant symbols. Since this is a hybrid form of error-correction coupled with error-detection feedback through the ARQ mechanism, it is commonly referred to as a hybrid automatic repeat request (HARQ) system.

The two known forms of HARQ are Chase combining and incremental redundancy (IR). Chase combining is a simplified form of HARQ, wherein the receiver simply requests retransmission of the original codeword and the received symbols corresponding to the codeword are combined together prior to repeated decoding and detection. IR provides for a transmission of additional parity symbols extending the length of the codeword and increasing the minimum Hamming distance, d_{min} between codewords. This results in a lower error rate following decoding of the extended codeword. The average throughput of such a system is higher than a fixed code rate system which always transmits codewords of maximum length and redundancy. In HARQ systems, it is a prerequisite that a reliable means be provided to detect errors in each decoded codeword. A system is described below which is able to provide an improvement to current HARQ systems by providing a more reliable means of error detection using the CRC and also provides for an improvement in

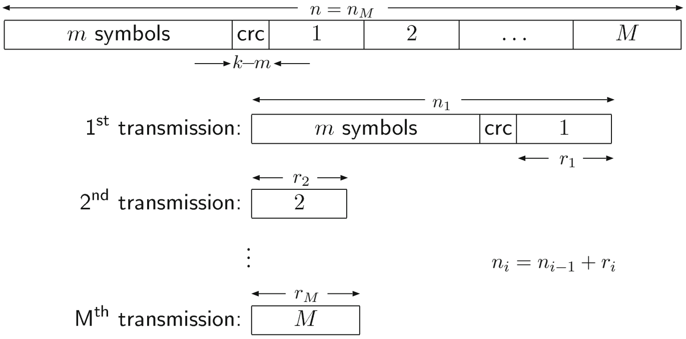

Fig. 17.1 Codeword format for conventional incremental-redundancy ARQ schemes

throughput by basing the error detection on the reliability of the detected codeword without the need to transmit the CRC.

Figure 17.1 shows the generic structure of the transmitted signal for a punctured codeword system. The transmitted signal comprises the initial codeword followed by additional parity symbols which are transmitted following each ARQ request up to a total of M transmissions for each codeword. All of the different types of codes used in HARQ systems: convolutional codes, turbo codes, LDPC codes, and algebraic codes can be constructed to fit into this generic codeword structure. As shown in Fig. 17.1, the maximum length of each codeword is n_M symbols transmitted in a total of M transmissions resulting from the reception of $M - 1$ negative ACK's (NACK's). The first transmission consists of m information symbols encoded into a total of n_1 symbols. There are r_1 parity symbols in addition to the CRC symbols. This is equivalent to puncturing the maximum length codeword in the last $n_M - n_1$ symbols. If this codeword is not decoded correctly, a NACK is received by the transmitter, (indicated either by the absence of an ACK being received or by a NACK signal being received), and r_2 parity symbols are transmitted as shown in Fig. 17.1.

The detection of an incorrect codeword is derived from the CRC in conventional HARQ systems. After the decoding of the received codeword, the CRC is recalculated and compared to the CRC symbols contained in the decoded codeword. If there is no match, then an incorrect codeword is declared and a NACK is conveyed to the transmitter. Following the second transmission, the decoder has a received codeword consisting of $n_1 + r_2$ symbols which are decoded. The CRC is recalculated and compared to the decoded CRC symbols. If there is still no match, a NACK is conveyed to the transmitter and the third transmission consists of the r_3 parity symbols and the net codeword consisting of $n_1 + r_2 + r_3$ symbols is decoded, and so on. The IR procedure ends either when an ACK is received by the transmitter or when a codeword of total length n_M symbols has been transmitted in a total of M transmissions.

Most conventional HARQ systems first encode the m information symbols plus CRC symbols into a codeword of length n_M symbols, where $\mathscr{C}_M = [n_M, k, d_M]$ denotes this code. The code $\mathscr{C}_M$ is then punctured by removing the last $n_M - n_{M-1}$

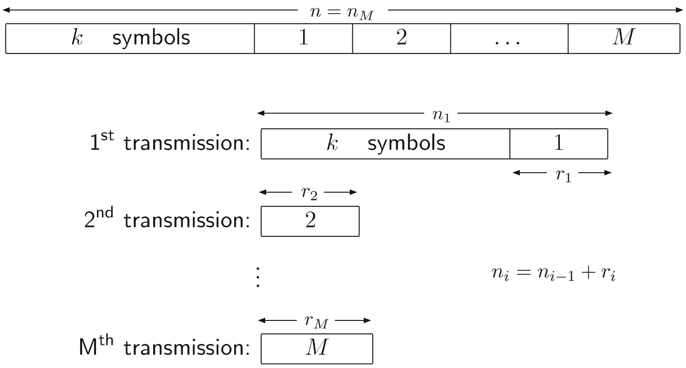

Fig. 17.2 Codeword format for the incremental-redundancy ARQ scheme without a CRC

symbols to produce a code $\mathscr{C}_{M-1} = [n_{M-1}, k, d_{M-1}]$, the code $\mathscr{C}_{M-1}$ is then punctured by removing the last $n_{M-1} - n_{M-2}$ symbols to produce a code $\mathscr{C}_{M-2}$, and so forth until a code $\mathscr{C}_1 = [n_1, k, d_1]$ is obtained. In this way, a sequence of codes $\mathscr{C}_1 = [n_1, k, d_1]$, $\mathscr{C}_2 = [n_2, k, d_2]$, ..., $\mathscr{C}_M = [n_M, k, d_M]$ is obtained. In the first transmission stage, a codeword $\mathscr{C}_1$ is transmitted, in the second transmission stage, the punctured parity symbols of $\mathscr{C}_2$ is transmitted and so on as shown in Fig. 17.1.

An alternative IR code construction method is to produce a sequence of codes using a generator matrix formed from a juxtaposition of the generator matrices of a nested block code. In this way, no puncturing is required.

Figure 17.2 shows the structure of the transmitted signal. The transmitted signal format is the same as Fig. 17.1 except that no CRC symbols are transmitted. The initial codeword consists only of the m information symbols plus the r_1 parity symbols. Additional parity symbols are transmitted following each ARQ request up to a total of M transmissions for each codeword. All of the different types of codes used in HARQ systems: convolutional codes, turbo codes, LDPC codes, and algebraic codes may be used in this format including the sequence of codes based on a nested block code construction.

Figure 17.3 shows a variation of the system where the k information symbols, denoted by vector $\boldsymbol{u}$, are encoded with the forward error-correction (FEC) encoder into n_M symbols denoted as $\boldsymbol{c}_M$ which are stored in the transmission controller. In the first transmission, n_1 symbols are transmitted. At the end of the ith stage, a codeword of total length n_i symbols has been transmitted. This corresponds to a codeword of length n_M symbols punctured in the last $n_M - n_i$ symbols. In Fig. 17.3, the codeword of length n_i is represented as a vector $\boldsymbol{v}$, which is then passed through the channel to produce $\boldsymbol{y}'$ and buffered in the Received buffer as $\boldsymbol{y}$ which is forward error-correction (FEC) decoded in the FEC decoder which produces the most likely codeword $\boldsymbol{c}_1$ and the next most likely codeword $\boldsymbol{c}_2$.

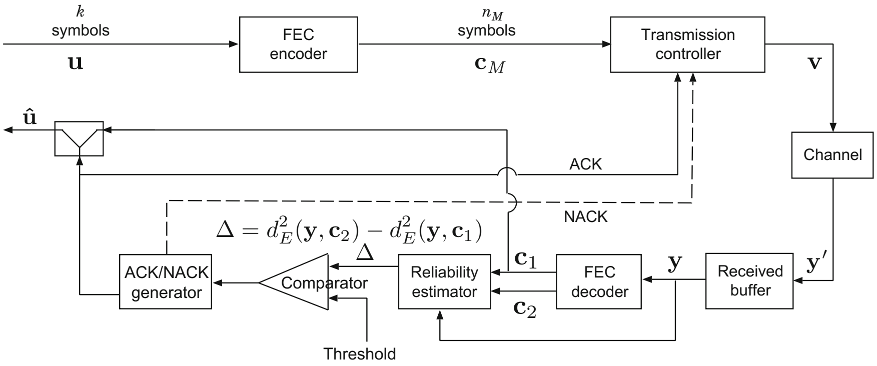

Fig. 17.3 The incremental-redundancy ARQ scheme with adjustable reliability without using a CRC

Let us consider that the IR system has had i transmissions so that a total of n_i symbols have been received and the total length of the transmitted codeword is n_i symbols.

$\boldsymbol{c}_1$ is denoted as

$$\boldsymbol{c}_1 = c_{10} + c_{11}x + c_{12}x^2 + c_{13}x^3 + c_{14}x^4 + \cdots + c_{1(n_i-1)}x^{n_i-1} \tag{17.8}$$

and $\boldsymbol{c}_2$ is denoted as

$$\boldsymbol{c}_2 = c_{20} + c_{21}x + c_{22}x^2 + c_{23}x^3 + c_{24}x^4 + \cdots + c_{2(n_i-1)}x^{n_i-1} \tag{17.9}$$

and the received symbols $\boldsymbol{y}$ are denoted as

$$\boldsymbol{y} = y_0 + y_1x + y_2x^2 + y_3x^3 + y_4x^4 + \cdots + y_{(n_i-1)2}x^{n_i-1} \tag{17.10}$$

For each decoded codeword, $\boldsymbol{c}_1$ and $\boldsymbol{c}_2$, the squared Euclidean distances $d_E^2(\boldsymbol{y}, \boldsymbol{c}_1)$ and $d_E^2(\boldsymbol{y}, \boldsymbol{c}_2)$ respectively are calculated between the codewords and the received symbols $\boldsymbol{y}$ stored in the Received buffer.

$d_E^2(\boldsymbol{y}, \boldsymbol{c}_1)$ is given by

$$d_E^2(\boldsymbol{y}, \boldsymbol{c}_1) = \sum_{j=0}^{n_i-1} (y_j - c_{1j})^2 \tag{17.11}$$

$d_E^2(\boldsymbol{y}, \boldsymbol{c}_2)$ is given by

$$d_E^2(\boldsymbol{y}, \boldsymbol{c}_2) = \sum_{j=0}^{n_i-1} (y_j - c_{2j})^2 \tag{17.12}$$

The function of the Reliability estimator shown in Fig. 17.3 is to determine how much smaller is $d_E^2(\boldsymbol{y}, \boldsymbol{c}_1)$ compared to $d_E^2(\boldsymbol{y}, \boldsymbol{c}_2)$ in order to estimate the likelihood that the codeword $\boldsymbol{c}_1$ is correct. The Reliability estimator calculates the squared Euclidean distances $d_E^2(\boldsymbol{y}, \boldsymbol{c}_1)$ and $d_E^2(\boldsymbol{y}, \boldsymbol{c}_2)$, and determines the difference Δ given by

$$\Delta = d_E^2(\boldsymbol{y}, \boldsymbol{c}_2) - d_E^2(\boldsymbol{y}, \boldsymbol{c}_1) \tag{17.13}$$

Δ is compared to a threshold which is calculated from the minimum Hamming distance of the first code in the sequence of codes, the absolute noise power, and a multiplicative constant, termed κ. As shown in Fig. 17.3, Δ is compared to the threshold by the Comparator. If Δ is not greater than the threshold, $\boldsymbol{c}_1$ is considered to be insufficiently reliable, and the output of the comparator causes the ACK/NACK generator to convey a NACK to the transmitter for more parity symbols to be transmitted. If Δ is greater than or equal to the threshold then $\boldsymbol{c}_1$ is considered to be correct, the output of the comparator causes the ACK/NACK generator to convey an ACK to the transmitter and in turn, the ACK/NACK generator causes the switch to close and $\boldsymbol{c}_1$ is switched to the output $\hat{\boldsymbol{u}}$. The ACK causes the entire IR procedure to begin again with a new vector $\boldsymbol{u}$. The way that Δ works as an indication of whether the codeword $\boldsymbol{c}_1$ is correct or not. If $\boldsymbol{c}_1$ is correct, then $d_E^2(\boldsymbol{y}, \boldsymbol{c}_1)$ is a summation of squared noise samples only because the signal terms cancel out. The codeword $\boldsymbol{c}_2$ differs from $\boldsymbol{c}_1$ in a number of symbol positions equal to at least the minimum Hamming distance of the current code, d_{min}. With the minimum squared Euclidean distance between symbols defined as d_S^2, Δ will be greater or equal to $d_{min} \times d_S^2$ plus a noise term dependent on the signal to noise ratio. If $\boldsymbol{c}_1$ is not correct $d_E^2(\boldsymbol{y}, \boldsymbol{c}_1)$ and $d_E^2(\boldsymbol{y}, \boldsymbol{c}_2)$ will be similar and Δ will be small.

If more parity symbols are transmitted because Δ is less than the threshold, the d_{min} of the code increases with each increase of codeword length and provided $\boldsymbol{c}_1$ is correct, Δ will increase accordingly.

The Reliability measure shown in Fig. 17.3 uses the squared Euclidean distance but it is apparent that equivalent soft decision metrics including cross-correlation and log likelihood may be used to the same effect.

In the system shown in Fig. 17.4 a CRC is transmitted in the first transmitted codeword. The m information symbols, shown as vector $\boldsymbol{u}$ in Fig. 17.4 are encoded with the CRC encoder to form a total of k symbols, shown as vector $\boldsymbol{x}$. The k symbols are encoded by the FEC encoder into n_M symbols denoted as $\boldsymbol{c}_M$ which are stored in the transmission controller. In the first transmission, n_1 symbols are transmitted. At the end of the ith stage, a codeword of total length n_i symbols has been transmitted. This corresponds to a codeword of length n_M symbols punctured in the last $n_M - n_i$ symbols. In Fig. 17.4, the codeword of length n_i is represented as a vector $\boldsymbol{v}$, which is then passed through the channel to produce $\boldsymbol{y}'$ and buffered in the Received buffer as $\boldsymbol{y}$, which is forward error-correction (FEC) decoded in the FEC decoder. The FEC decoder produces L codewords with decreasing reliability as measured by the squared Euclidean distance between each codeword and the received symbols or as measured by an equivalent soft decision metric such as cross-

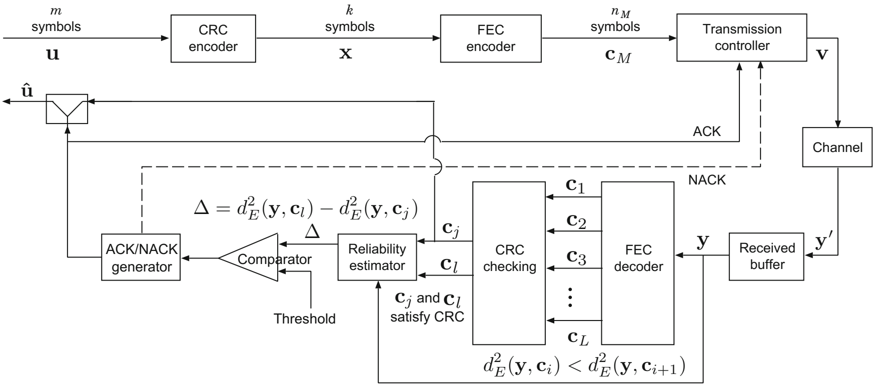

Fig. 17.4 The incremental-redundancy ARQ scheme with adjustable reliability using a CRC

correlation between each codeword and the received symbols. The L codewords are input to CRC checking which determines the most reliable codeword, $\boldsymbol{c}_j$, which satisfies the CRC and the next most reliable codeword, $\boldsymbol{c}_l$, which satisfies the CRC. The Reliability estimator shown in Fig. 17.4 determines the difference, Δ, of the squared Euclidean distances between codewords $\boldsymbol{c}_j$ and $\boldsymbol{c}_l$ and the corresponding received symbols.

Δ is given by

$$\Delta = d_E^2(\boldsymbol{y}, \boldsymbol{c}_l) - d_E^2(\boldsymbol{y}, \boldsymbol{c}_j) \tag{17.14}$$

Δ is compared to a threshold which is calculated from the minimum Hamming distance of the first code in the sequence of codes, the absolute noise power, and a multiplicative constant termed κ. As shown in Fig. 17.4, Δ is compared to the threshold by the comparator. If Δ is not greater than the threshold, $\boldsymbol{c}_j$ is considered to be insufficiently reliable, and the output of the comparator causes the ACK/NACK generator to convey a NACK to the transmitter for more parity symbols to be transmitted. If Δ is greater than or equal to the threshold then $\boldsymbol{c}_j$ is considered to be correct, the output of the comparator causes the ACK/NACK generator to convey an ACK to the transmitter and in turn, the ACK/NACK generator causes the switch to close and $\boldsymbol{c}_j$ is switched to the output $\hat{\boldsymbol{u}}$. The ACK causes the entire IR procedure to begin again with a new vector $\boldsymbol{u}$.

The Reliability measure shown in Fig. 17.4 uses the squared Euclidean distance but it is apparent that equivalent soft decision metrics including cross correlation and log likelihood ratios may be used to the same effect.

17.2.1.1 Code Generation Using Nested Block Codes

If $\mathscr{C}$ is a cyclic code, then there exists a generator polynomial $g(x) \in \mathbb{F}_2[x]$ and a parity-check polynomial $h(x) \in \mathbb{F}_2[x]$ such that $g(x)h(x) = x^{n_1} - 1$. Two cyclic codes, $\mathscr{C}_1$ with $g_1(x)$ as the generator polynomial and $\mathscr{C}_2$ with $g_2(x)$ as the generator polynomial, are said to be chained or nested, if $g_1(x)|g_2(x)$, and we denote them by $\mathscr{C}_1 \supset \mathscr{C}_2$. With reference to this definition, it is clear that narrow-sense BCH codes of the same length form a chain of cyclic codes. Given a chain of two codes, using a code construction method known as Construction X, a construction method first described by Sloane et al. [5], the code with larger dimension can be lengthened to produce a code with increased length and minimum distance.

A generalised form of Construction X involves more than two codes. Let $\mathscr{B}_i$ be an $[n_1, k_i, d_i]$ code, given a chain of M codes, $\mathscr{B}_1 \supset \mathscr{B}_2 \supset \cdots \supset \mathscr{B}_M$, and a set of auxiliary codes $\mathscr{A}_i = [n'_i, k'_i, d'_i]$, for $1 \le i \le M-1$, where $k'_i = k_1 - k_i$, a code $\mathscr{C}_X = [n_1 + \sum_{i=1}^{M-1} n'_i, k_1, d]$ can be constructed, where $d = \min\{d_M, d_{M-1} + d'_{M-1}, d_{M-2} + d'_{M-2} + d'_{M-1}, \ldots, d_1 + \sum_{i=1}^{M-1} d'_i\}$.

Denoting z as a vector of length n_1 formed by the first n_1 coordinates of a codeword of $\mathscr{C}_X$. A codeword of $\mathscr{C}_X$ is a juxtaposition of codewords of $\mathscr{B}_i$ and $\mathscr{A}_i$, where

$$\begin{array}{lcl}
(\ \boldsymbol{b}_M \mid \boldsymbol{0} \mid \boldsymbol{0} \mid \ldots \mid \boldsymbol{0} \mid \boldsymbol{0}\ ) & & \text{if } z \in \mathscr{B}_M,\\
(\ \boldsymbol{b}_{M-1} \mid \boldsymbol{0} \mid \boldsymbol{0} \mid \ldots \mid \boldsymbol{0} \mid \boldsymbol{a}_{M-1}\ ) & & \text{if } z \in \mathscr{B}_{M-1},\\
(\ \boldsymbol{b}_{M-2} \mid \boldsymbol{0} \mid \boldsymbol{0} \mid \ldots \mid \boldsymbol{a}_{M-2} \mid \boldsymbol{a}_{M-1}\ ) & & \text{if } z \in \mathscr{B}_{M-2},\\
\vdots & & \vdots\\
(\ \boldsymbol{b}_2 \mid \boldsymbol{0} \mid \boldsymbol{a}_2 \mid \ldots \mid \boldsymbol{a}_{M-2} \mid \boldsymbol{a}_{M-1}\ ) & & \text{if } z \in \mathscr{B}_2,\\
(\ \boldsymbol{b}_1 \mid \boldsymbol{a}_1 \mid \boldsymbol{a}_2 \mid \ldots \mid \boldsymbol{a}_{M-2} \mid \boldsymbol{a}_{M-1}\ ) & & \text{if } z \in \mathscr{B}_1,
\end{array}$$

where $\boldsymbol{b}_i \in \mathscr{B}_i$ and $\boldsymbol{a}_i \in \mathscr{A}_i$.

17.2.1.2 Example of Code Generation Using Nested Block Codes

There exists a chain of extended BCH codes of length 128 bits,

$$\mathscr{B}_1 = [128, 113, 6] \supset \mathscr{B}_2 = [128, 92, 12] \supset \mathscr{B}_3 = [128, 78, 16] \supset \\ \mathscr{B}_4 = [128, 71, 20].$$

Applying Construction X to $[128, 113, 6] \supset [128, 92, 12]$ with an $[32, 21, 6]$ extended BCH code as auxiliary code, a $[160, 113, 12]$ code is obtained, giving

$$[160, 113, 12] \supset [160, 92, 12] \supset [160, 78, 16] \supset [160, 71, 20].$$

Additionally, using a $[42, 35, 4]$ shortened extended Hamming code as the auxiliary code in applying Construction X to $[160, 113, 12] \supset [160, 78, 16]$, giving

$$[202, 113, 16] \supset [202, 92, 16] \supset [202, 78, 16] \supset [202, 71, 20].$$

Finally, applying Construction X to [202, 113, 16] ⊃ [202, 71, 20] with the shortened extended Hamming code [49, 42, 4] as the auxiliary code, giving

$$[251, 113, 20] \supset [251, 92, 20] \supset [251, 78, 20] \supset [251, 71, 20].$$

The resulting sequence of codes which are used in this example are [128, 113, 6], [160, 113, 12], [202, 113, 16] and [251, 113, 20].

The generator matrix of the last code, the [251, 113, 20] code is given by

$$\mathbf{G} = \left(\begin{array}{c|cccc||c|c|c}
\mathbf{I}_{71} & & -\mathbf{R}_4 & & & 0 & 0 & 0 \\ \hline
 & \mathbf{I}_7 & -\mathbf{R}_3 & & & & & \mathbf{G}_{\mathscr{A}_3} \\
\mathbf{0} & & \mathbf{I}_{14} & -\mathbf{R}_2 & & & \mathbf{G}_{\mathscr{A}_2} & \\
 & & & \mathbf{I}_{21} & -\mathbf{R}_1 & \mathbf{G}_{\mathscr{A}_1} & &
\end{array}\right). \tag{17.15}$$

On the left hand side of the double bar, the generator matrix of the code $\mathscr{B}_1$ is decomposed along the chain $\mathscr{B}_1 \supset \mathscr{B}_2 \supset \mathscr{B}_3 \supset \mathscr{B}_4$. The matrices $\mathbf{G}_{\mathscr{A}_i}$, for $1 \le i \le 3$ are the generator matrices of the auxiliary codes $\mathscr{A}_i$.

This generator matrix is used to generate each entire codeword of length $n_M = 251$ bits, but these bits are not transmitted unless requested. The first 128 bits of each entire codeword are selected to form the codeword of the code [128, 113, 6] and are transmitted first, bit 0 through to bit 127. The next transmission (if requested by the IR system) consists of 32 parity bits. These are bit 128 through to bit 159 of the entire codeword. These 32 parity bits plus the original 128 bits form a codeword of the [160, 113, 12] code. The next transmission (if requested by the IR system) consists of 42 parity bits. These are bit 160 through to bit 201 of the entire codeword. These 42 parity bits plus the previously transmitted 160 bits form a codeword from the [202, 113, 16] code. The last transmission (if requested by the IR system) consists of 49 parity bits. These are the last 49 bits, bit 202 through to bit 250, of the entire codeword. These 49 parity bits plus the previously transmitted 202 bits form a codeword from the [251, 113, 20] code. The sequence of increasing length codewords with each transmission (if requested by the IR system) has a minimum Hamming distance which starts with 6, increases from 6 to 12, then to 16 and finally, to 20. In turn this will produce an increasing reliability given by Eq. (17.13) or (17.14) depending on the type of system.

A completely different method of generating nested codes is to use the external parity checks, augmentation method first suggested by Goppa in which independent columns are added incrementally to the parity-check matrix. The method is described in detail in Chap. 6 and can be applied to any Goppa or BCH code.

In order to be used in the HARQ systems, a FEC decoder is needed that will decode these nested block codes. One such universal decoder is the modified Dorsch decoder described in Chap. 15 and results using this decoder are presented below.

17.2.1.3 List Decoder for Turbo and LDPC Codes

If LDPC or turbo codes are to be used, the HARQ system needs a decoder that provides several codewords at its output in order that the difference between the squared Euclidean distances (or an equivalent soft decision metric) of the most likely transmitted codeword and the next most likely transmitted codeword may be determined and compared to the threshold. For turbo codes, the conventional decoder is not a list decoder but Narayanan and Stuber [3] show how a list decoder may be provided for turbo codes. Similarly for LDPC codes, Kristensen [1] shows how a list decoder may be provided for LDPC codes.

17.2.1.4 Performance Results Using the Nested Codes

Computer simulations using the nested codes constructed above have been carried out featuring all three HARQ systems. These systems include the traditional HARQ system using hard decision checks of the CRC and the two new systems featuring the soft decision, decoded codeword/received vector check, with or without a CRC. All of the simulations of the three systems have been carried out using a modified Dorsch decoder as described in Chap. 15. The modified Dorsch decoder can be easily configured as a list decoder with hard and soft decision outputs.

For each one of the nested codes, the decoder exhibits almost optimum maximum likelihood performance by virtue of its delta correlation algorithm corresponding to a total of 10^6 codewords, that are closest to the received vector, being evaluated each time there is a new received vector to input. Since the decoder knows which of the nested codes it is decoding, it is possible to optimise the settings of the decoder for each code.

For the CRC cases, an 8 bit CRC polynomial $(1 + x)(1 + x^2 + x^5 + x^6 + x^7)$ was used, the 8 CRC bits being included in each codeword. It should be noted that in calculating the throughput these CRC bits are not counted as information bits. In the CRC cases, there are 105 information bits per transmitted codeword. In the computer simulations, an ACK is transmitted if Δ is greater than threshold or there have been M IR transmissions, otherwise a NACK is transmitted.

The traditional HARQ system using a CRC is compared to the new system not using a CRC in Figs. 17.5 and 17.6. The comparative frame error rate (FER) performance is shown in Fig. 17.5 and the throughput is shown in Fig. 17.6 as a function

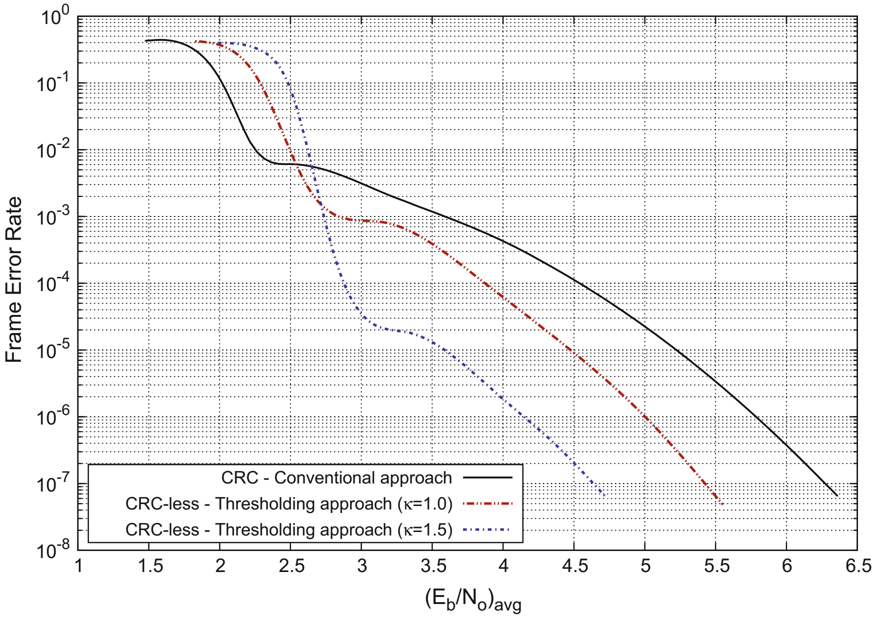

Fig. 17.5 The error rate performance in comparison to the classical HARQ scheme using a CRC

of the average $\frac{E_b}{N_o}$ ratio. The traditional CRC approach shows good throughput, but exhibits an early error-floor of the FER, which is caused by undetected error events. The FER performance shows the benefit of having increased reliability of error detection compared to the traditional CRC approach. Two threshold settings are provided using the multiplicative constant κ and the effects of these are shown in Figs. 17.5 and 17.6. It is apparent from the graphs that the threshold setting may be used to trade-off throughput against reduced FER. The improvements in both throughput and FER provided by the new HARQ systems compared to the conventional HARQ system, featuring a hard decision CRC check, are evident from Figs. 17.5 and 17.6.

The comparative FER performance and throughput with a CRC compared to not using a CRC is shown in Figs. 17.7 and 17.8 for the new system where the threshold is fixed by $\kappa = 1$. The new system using a CRC shows an improvement in FER, Fig. 17.7, over the entire range of average $\frac{E_b}{N_o}$ and an improvement in throughput, Fig. 17.8, also over the entire range of average $\frac{E_b}{N_o}$ compared to the traditional HARQ approach using a CRC.

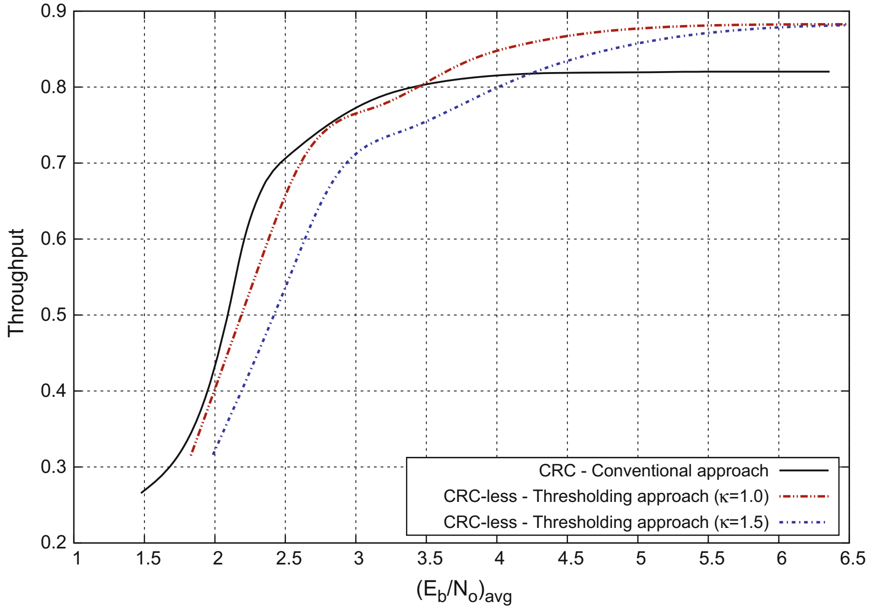

Fig. 17.6 The throughput performance without using a CRC in comparison to the classical HARQ scheme using a CRC

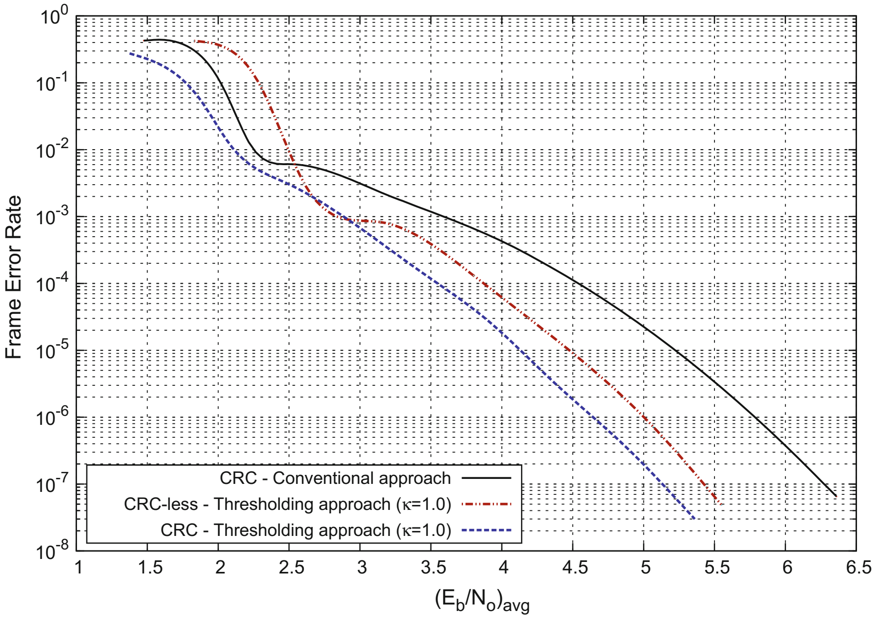

Fig. 17.7 The error rate performance using a CRC in comparison to the classical HARQ scheme using a CRC

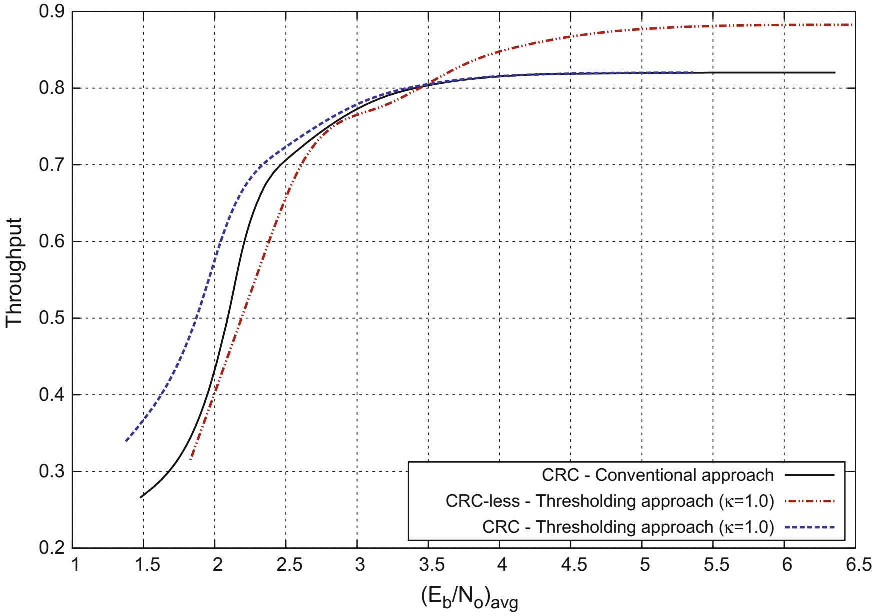

Fig. 17.8 The throughput performance with a CRC in comparison to the classical HARQ scheme using a CRC

17.3 Summary

This chapter has discussed the design of codes and systems for combined error detection and correction, primarily aimed at applications featuring retransmission of data packets which have not been decoded correctly. Several such Hybrid Automatic ReQuest, HARQ, systems have been described including a novel system variation which uses a retransmission metric based on a soft decision; the Euclidean distance between the decoded codeword and the received vector. It has been shown that a cyclic redundancy check, CRC, is not essential for this system and need not be transmitted.

It has also been shown how to construct the generator matrix of a nested set of block codes of length 251 bits by applying Construction X three times in succession starting with an extended BCH (128, 113, 6) code. The resulting nested codes have been used as the basis for an incremental-redundancy system whereby the first 128 bits transmitted is a codeword from the BCH code, followed by the transmission of a further 32 bits, if requested, producing a codeword from a (160, 113, 12) code. Further requests for additional transmitted bits finally result in a codeword from a (251, 113, 20) code, each time increasing the chance of correct decoding by increasing the minimum Hamming distance of the net received codeword. Performance graphs have been presented showing the comparative error rate performances and throughputs of the new HARQ systems compared to the standard HARQ system. The advantages of lower error floors and increased throughputs are evident from the presented graphs.

References

1. Kristensen J.T.: List Decoding of LDPC Codes, Masters thesis 2007-02, Technical University of Denmark
2. MacWilliams, F.J., Sloane, N.J.A.: The Theory of Error-Correcting Codes. North-Holland, Amsterdam (1977)
3. Narayanan, K.R., Stuber, G.L.: List decoding of Turbo codes. IEEE Trans. Commun. **46**(6), 754–762 (1998)
4. Peterson, W.W., Weldon Jr., E.J.: Error-Correcting Codes. MIT Press, Cambridge (1972)
5. Sloane, N.J., Reddy, S.M., Chen, C.L.: New binary codes. IEEE Trans. Inf. Theory **18**(4), 503–510 (1972)