Chapter 2
Soft and Hard Decision Decoding Performance

2.1 Introduction

This chapter is concerned with the performance of binary codes under maximum likelihood soft decision decoding and maximum likelihood hard decision decoding. Maximum likelihood decoding gives the best performance possible for a code and is therefore used to assess the quality of the code. In practice, maximum likelihood decoding of codes is computationally difficult, and as such, theoretical bounds on the performance of codes are used instead. These bounds are in lower and upper form and the expected performance of the code is within the region bounded by the two. For hard decision decoding, lower and upper bounds on maximum likelihood decoding are computed using information on the coset weight leader distribution. For maximum likelihood soft decision decoding, the bounds are computed using the weight distribution of the codes. The union bound is a simple and well-known bound for the performance of codes under maximum likelihood soft decision decoding. The union bound can be expressed as both an upper and lower bound. Using these bounds, we see that as the SNR per bit becomes large the performance of the codes can be completely determined by the lower bound. However, this is not the case with the bounds on maximum likelihood hard decision decoding of codes. In general, soft decision decoding has better performance than hard decision decoding and being able to estimate the performance of codes under soft decision decoding is attractive. Computation of the union bound requires the knowledge of the weight distribution of the code. In Sect. 2.3.1, we use a binomial approximation for the weight distribution of codes for which the actual computation of the weight distribution is prohibitive. As a result, it possible to calculate within an acceptable degree of error the region in which the performance of codes can be completely predicted.

2.2 Hard Decision Performance

2.2.1 Complete and Bounded Distance Decoding

Hard decision decoding is concerned with decoding of the received sequence in
hamming space. Typically, the real-valued received sequence is quantised using a
threshold to a binary sequence. A bounded distance decoder is guaranteed to correct
all t errors or less, where t is called the packing radius and is given by:

$$t = \left\lfloor \frac{d-1}{2} \right\rfloor$$

and d is the minimum hamming distance of the code. Within a sphere centred
around a codeword in the hamming space of radius t there is no other codeword,
and the received sequence in this sphere is closest to the codeword. Beyond the
packing radius, some error patterns may be corrected. A complete decoder exhaus-
tively matches all codewords to the received sequence and selects the codeword with
minimum hamming distance. A complete decoder is also called a minimum dis-
tance decoder or maximum likelihood decoder. Thus, a complete decoder corrects
some patterns of error beyond the packing radius. The complexity of implementing
a complete decoder is known to be NP-complete [3]. Complete decoding can be
accomplished using a standard array. In order to discuss standard array decoding, we
first need to define cosets and coset leaders.

Definition 2.1 A coset of a code $\mathscr{C}$ is a set containing all the codewords of $\mathscr{C}$
corrupted by a single sequence $\mathbf{a} \in \mathbb{F}_q^n \setminus \mathscr{C} \cup \{\mathbf{0}\}$.

A coset of a binary code contains 2^k sequences and there are 2^{n-k} possible cosets.
Any sequence of minimum hamming weight in a coset can be chosen as a coset
leader. In order to use a standard array, the coset leaders of all the cosets of a code
must be known. We illustrate complete decoding with an example. Using a $(7, 3)$
dual Hamming code with the following generator matrix

$$G = \begin{bmatrix} 1\,0\,0\,0\,1\,1\,1 \\ 0\,1\,0\,1\,0\,1\,1 \\ 0\,0\,1\,1\,1\,0\,1 \end{bmatrix}$$

This code has codewords

$$C = \left\{ \begin{matrix} 0\,0\,0\,0\,0\,0\,0 \\ 1\,0\,0\,0\,1\,1\,1 \\ 0\,1\,0\,1\,0\,1\,1 \\ 0\,0\,1\,1\,1\,0\,1 \\ 1\,1\,0\,1\,1\,0\,0 \\ 0\,1\,1\,0\,1\,1\,0 \\ 1\,0\,1\,1\,0\,1\,0 \\ 1\,1\,1\,0\,0\,0\,1 \end{matrix} \right\}$$

Coset Leaders							
0000000	**1000111**	**0101011**	**0011101**	**1101100**	**0110110**	**1011010**	**1110001**
0000001	1000110	0101010	0011100	1101101	0110111	1011011	1110000
0000010	1000101	0101001	0011111	1101110	0110100	1011000	1110011
0000100	1000011	0101111	0011001	1101000	0110010	1011110	1110101
0001000	1001111	0100011	0010101	1100100	0111110	1010010	1111001
0010000	1010111	0111011	0001101	1111100	0111110	1001010	1100001
0100000	1100111	0001011	0111101	1001100	0010110	1111010	1010001
1000000	0000111	1101011	1011101	0101100	1110110	0011010	0110001
0000011	1000100	0101000	0011110	1101111	0110101	1011001	1110010
0000110	1000001	0101101	0011011	1101010	0110000	1011100	1110111
0001100	1001011	0100111	0010001	1100000	0111010	1010110	1111101
0011000	1011111	0110011	0000101	1110100	0101110	1000010	1101001
0001010	1001101	0100001	0010111	1100110	0111100	1010000	1111011
0010100	1010011	0111111	0001001	1111000	0100010	1001110	1100101
0010010	1010101	0111001	0001111	1111110	0100100	1001000	1100011
0001110	1001001	0100101	0010011	1100010	0111000	1010100	1111111

Fig. 2.1 Standard array for the (7, 3, 4) binary code

Complete decoding can be accomplished using standard array decoding. The example code is decoded using standard array decoding as follows, The top row of the array in Fig. 2.1 in bold contains the codewords of the (7, 3, 4) code.[1] Subsequent rows contain all the other cosets of the code with the array arranged so that the coset leaders are in the first column. The decoder finds the received sequence on a row in the array and then subtracts the coset leader corresponding to that row from it to obtain a decoded sequence. The standard array is partitioned based on the weight of the coset leaders. Received sequences on rows with coset leaders of weight less than or equal to $t = \frac{3-1}{2} = 1$ are all corrected. Some received sequences on rows with coset leaders with weight greater than t are also corrected. Examining the standard array, it can be seen that the code can correct all single error sequences, some two error sequences and one three error sequence. The coset weight $\mathbb{C}_i$ distribution is

$$\mathbb{C}_0 = 1$$
$$\mathbb{C}_1 = 7$$
$$\mathbb{C}_2 = 7$$
$$\mathbb{C}_3 = 1$$

The covering radius of the code is the weight of the largest coset leader (in this example it is 3).

---

[1]It is worth noting that a code itself can be considered as a coset with the sequence **a** an all zero sequence.

2.2.2 The Performance of Codes on the Binary Symmetric Channel

Consider a real-valued sequence received from a transmission through an AWGN channel. If a demodulator makes hard decisions at the receiver, the channel may be modelled as a binary symmetric channel. Assuming the probability of bit error for the BSC is p, the probability of decoding error with a bounded distance decoder is given by,

$$P_{\text{BDD}}(e) = 1 - \sum_{i=0}^{t} \mathbb{C}_i \, p^i (1-p)^{n-i} \tag{2.1}$$

where $\mathbb{C}_i$ is the number of coset leaders with weight i. $\mathbb{C}_i$ known for $0 \le i \le t$ and is given by,

$$\mathbb{C}_i = \binom{n}{i} \quad 0 \le i \le t.$$

However, $\mathbb{C}_i$, $i > t$ need to be computed for individual codes. The probability of error after full decoding is

$$P_{\text{Full}}(e) = 1 - \sum_{i=0}^{n} \mathbb{C}_i \, p^i (1-p)^{n-i}. \tag{2.2}$$

Figure 2.2 shows the performance of the bounded distance decoder and the full decoder for different codes. The bounds are computed using (2.1) and (2.2). As expected, there is significant coding gain between unencoded and coded transmission (bounded distance and full decoding) for all the cases. There is a small coding gain between bounded distance and full decoders. This coding gain depends on the coset leader weight distribution $\mathbb{C}_i$ for $i > t$ of the individual codes. The balance between complexity and performance for full and bounded distance decoders[2] ensures that the latter are preferred in practice. Observe that in Fig. 2.2 that the complete decoder consistently outperforms the bounded distance decoder as the probability of error decreases and $\frac{E_b}{N_0}$ increases. We will see in Sect. 2.3 that a similar setup using soft decision decoding in Euclidean space produces different results.

2.2.2.1 Bounds on Decoding on the BSC Channel

Suppose s is such that $\mathbb{C}_s$ is the maximum non-zero value for a code then s is the covering radius of the code. If the covering radius s of a code is known and $\mathbb{C}_i$, $i > t$ are not known, then the probability of error after decoding can be bounded by

---

[2]Bounded distance decoders usually have polynomial complexity, e.g. the Berlekamp Massey decoder for BCH codes has complexity $O(t^2)$ [1].

(a) BCH Code (63,36)

(b) BCH Code (63,39)

(c) Goppa Code (128,100)

(d) BCH Code (127,92)
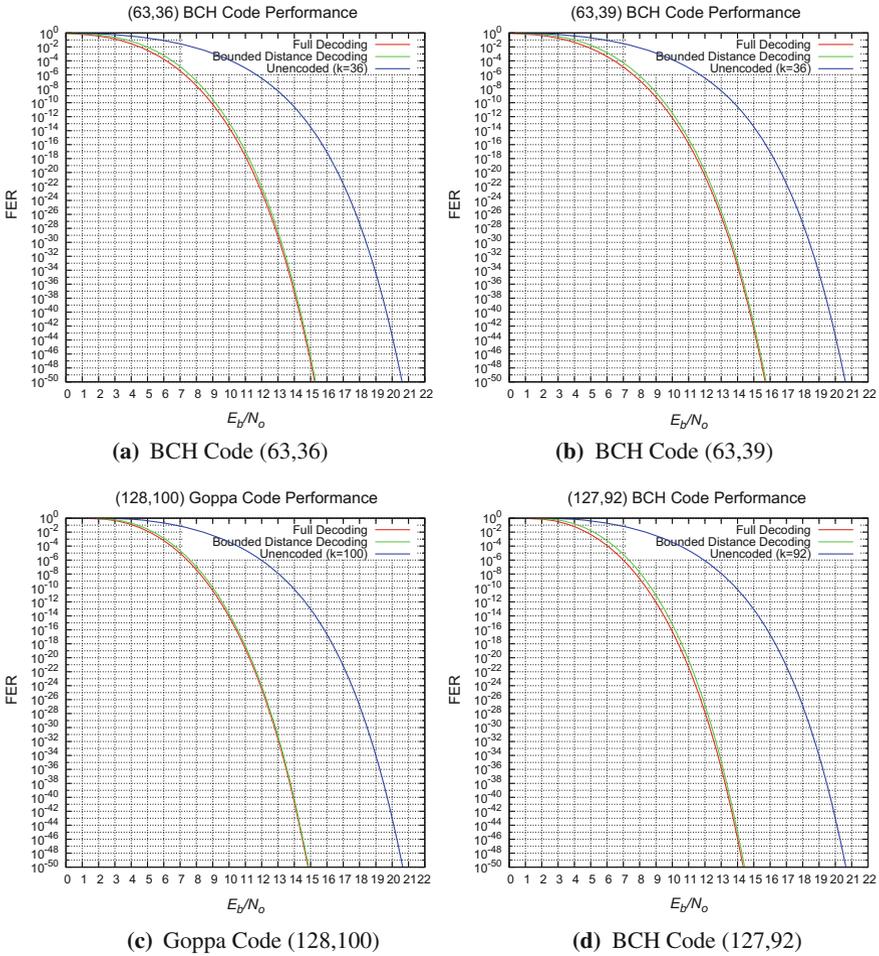
Fig. 2.2 BCH code BDD and full decoder performance, frame error rate (FER) against $\frac{E_b}{N_0}$

$$P_e \geq 1 - \left[\sum_{i=0}^{t} \binom{n}{i} p^i (1-p)^{n-i} + p^s (1-p)^{n-s} \right] \tag{2.3}$$

$$\leq 1 - \left[\sum_{i=0}^{t} \binom{n}{i} p^i (1-p)^{n-i} + \mathbb{W}_s p^s (1-p)^{n-s} \right] \tag{2.4}$$

assuming the code can correct t errors and

$$\mathbb{W}_s = 2^{n-k} - \sum_{i=0}^{t} \binom{n}{i}.$$

The lower bound assumes that there is a single coset leader of weight s, and hence the term $p^s(1-p)^{n-s}$ while the upper bound assumes that all the coset leaders of weight greater than t have weight equal to the covering radius s. For the lower bound to hold, $\mathbb{W}_s \geq 1$. The lower bound can be further tightened by assuming that the $\mathbb{W}_s - 1$ cosets have weight of $t+1, t+2, \ldots$ until they can all be accounted for.[3]

2.3 Soft Decision Performance

The union bound for the probability of sequence error using maximum likelihood soft decoding performance on binary codes with BPSK modulation in the AWGN channel is given by [2],

$$P_s \leq \frac{1}{2} \sum_{j=1}^{n} A_j \, \mathrm{erfc}\left(\sqrt{\frac{E_b}{N_0}Rj}\right) \tag{2.5}$$

where R is the code rate, A_j is the number of codewords of weight j and $\frac{E_b}{N_0}$ is the SNR per bit. The union bound is obtained by assuming that events in which the received sequence is closer in euclidean distance to a codeword of weight j are independent as such the probability of error is the sum of all these events. A drawback to the exact computation of the union bound is the fact that the weight distribution A_j, $0 \leq j \leq n$ of the code is required. Except for a small number of cases, the complete weight distribution of many codes is not known due to complexity limitations. Since $A_j = 0$ for $1 \leq j < d$ where d is the minimum distance of the code we can express (2.5) as,

$$P_s \leq \frac{1}{2} \sum_{j=d}^{n} A_j \, \mathrm{erfc}\left(\sqrt{\frac{E_b}{N_0}Rj}\right) \tag{2.6}$$

$$\leq \frac{1}{2} A_d \, \mathrm{erfc}\left(\sqrt{\frac{E_b}{N_0}Rd}\right) + \frac{1}{2} \sum_{j=d+1}^{n} A_j \, \mathrm{erfc}\left(\sqrt{\frac{E_b}{N_0}Rj}\right) \tag{2.7}$$

A lower bound on the probability of error can be obtained if it is assumed that error events occur only when the received sequence is closer in euclidean distance to codewords at a distance d from the correct codeword.

$$P_s \geq \frac{1}{2} A_d \, \mathrm{erfc}\left(\sqrt{\frac{E_b}{N_0}Rd}\right) \tag{2.8}$$

---

[3]This can be viewed as the code only has one term at the covering radius, and all other terms are at $t+1$.

where

$$\frac{1}{2} \sum_{j=d+1}^{n} A_j \, \text{erfc}\left(\sqrt{\frac{E_b}{N_0}Rj}\right) = 0. \tag{2.9}$$

As such,

$$\frac{1}{2} A_d \, \text{erfc}\left(\sqrt{\frac{E_b}{N_0}Rd}\right) \leq P_s \leq \frac{1}{2} \sum_{j=d}^{n} A_j \, \text{erfc}\left(\sqrt{\frac{E_b}{N_0}Rj}\right) \tag{2.10}$$

Therefore, the practical soft decision performance of a binary code lies between the upper and lower Union bound. It will be instructive to observe the union bound performance for actual codes using their computed weight distributions as the SNR per bit $\frac{E_b}{N_0}$ increases. By allowing $\frac{E_b}{N_0}$ to become large (and P_s to decrease) simulations for several codes suggest that at a certain *intersection* value of $\frac{E_b}{N_0}$ the upper bound equals the lower bound. Consider Figs. 2.3, 2.4 and 2.5 which show the frame error rate against the SNR per bit for three types of codes. The upper bounds in the figures are obtained using the complete weight distribution of the codes with Eq. (2.5). The lower bounds are obtained using only the number of codewords of minimum weight of the codes with Eq. (2.8). It can be observed that as $\frac{E_b}{N_0}$ becomes large, the upper bound meets and equals the lower bound. The significance of this observation is that for $\frac{E_b}{N_0}$ values above the point where the two bounds intersect the performance of the codes under soft decision can be completely determined by the lower bound (or the upper bound). In this region where the bounds agree, when errors occur they do so because the received sequence is closer to codewords a distance d away from the correct codeword. The actual performance of the codes before this region is somewhere between the upper and lower bounds. As we have seen earlier, the two bounds agree when the sum in (2.9) approaches 0. It may be useful to consider an approximation of the complementary error function (erfc),

$$\text{erfc}(x) < e^{-x^2}$$

in which case the condition becomes

$$\frac{1}{2} \sum_{j=d+1}^{n} A_j \, e^{-\frac{E_b}{N_0}Rj} \approx 0. \tag{2.11}$$

Clearly, the sum approximates to zero if each term in the sum also approximates to zero. It is safe to assume that the term $A_j \, \text{erfc}\left(\sqrt{\frac{E_b}{N_0}Rj}\right)$ decreases as j increases since $\text{erfc}\left(\sqrt{\frac{E_b}{N_0}Rj}\right)$ reduces exponentially with j and A_j increases in a binomial (in most cases). The size of the gap between the lower and upper bounds is also

(a) Extended BCH Code (128,29)

(b) Extended BCH Code (128,64)

(c) Extended BCH Code (128,85)

(d) Extended BCH Code (128,120)
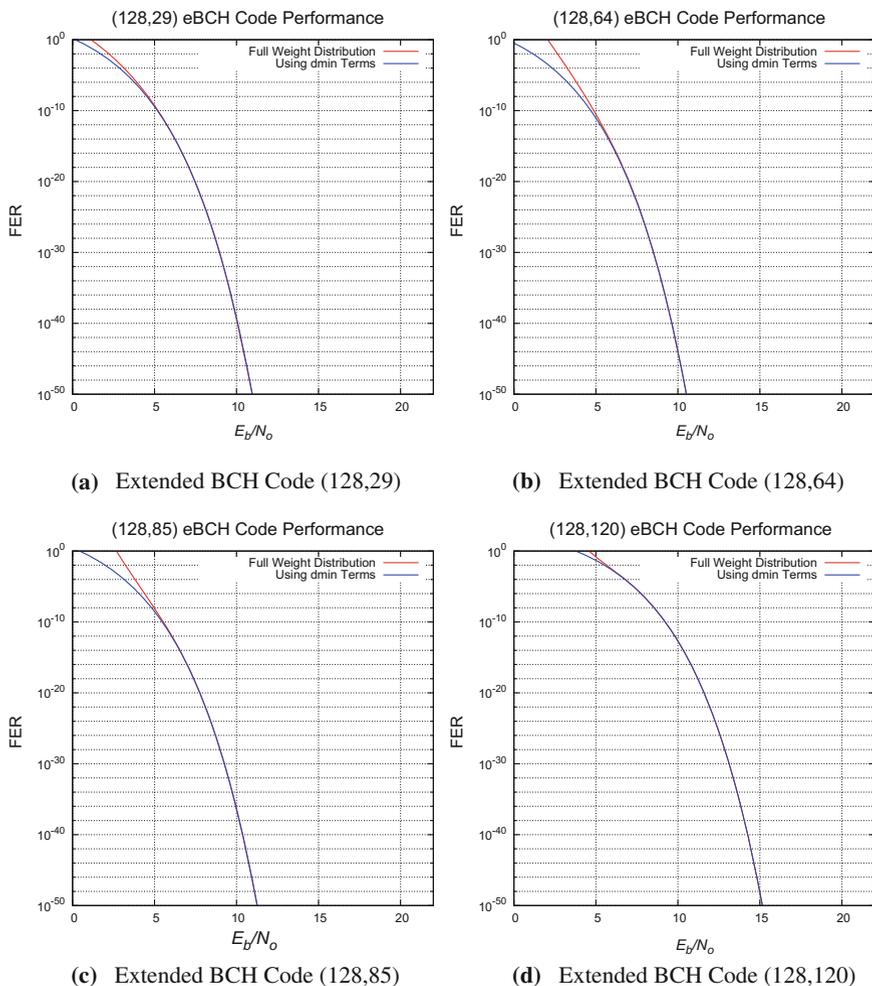
Fig. 2.3 Extended BCH code lower and upper union bound performance, frame error rate (FER) against $\frac{E_b}{N_0}$

determined by these terms. Each term $A_j\,e^{-\frac{E_b}{N_0}Rj}$ becomes small if one or both of the following conditions are met,

(a) Some of the A_j, $j > d$ are zero. This is common in low rate binary codes with a small number of codewords.
(b) The product $\frac{E_b}{N_0}Rj$ for $j > d$ becomes very large.

Observing Fig. 2.3, 2.4 and 2.5, it can be seen that at small values of $\frac{E_b}{N_0}$ and for low rate codes for which $R = \frac{k}{n}$ is small have some $A_j = 0$, $j > d$ and as such the gaps
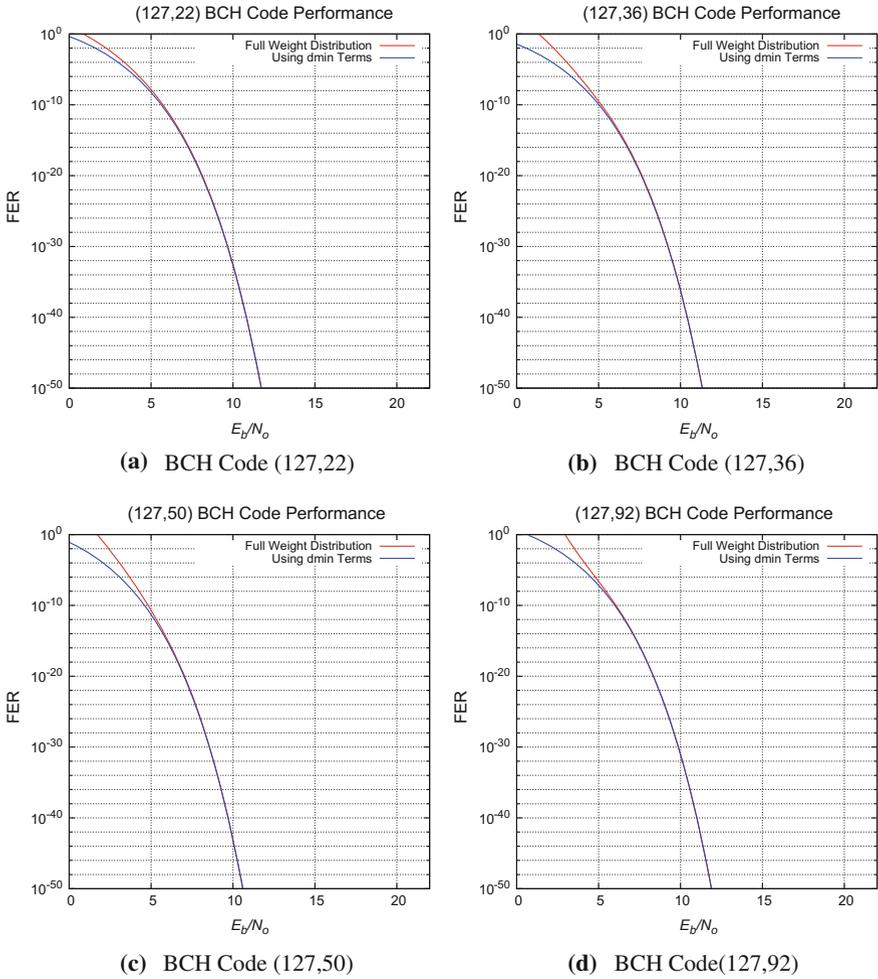
Fig. 2.4 BCH code lower and upper union bound performance, frame error rate (FER) against $\frac{E_b}{N_0}$

between the upper and lower bounds are small. As an example consider the low rate (127, 22, 47) BCH code in Fig. 2.4a which has,

$$A_j = 0 \quad j \in \{49\ldots54\} \cup \{57\ldots62\} \cup \{65\ldots70\} \cup \{73\ldots78\} \cup \{81\ldots126\}.$$

For the high rate codes, R is large so that the product $\frac{E_b}{N_0}Rj$ becomes very large therefore the gaps between the upper and lower bounds are small.

Figure 2.6 compares bounded distance decoding and full decoding with maximum likelihood soft decision decoding of the (63, 39) and (63, 36) BCH codes. It can be seen from the figure that whilst the probability of error for maximum likelihood
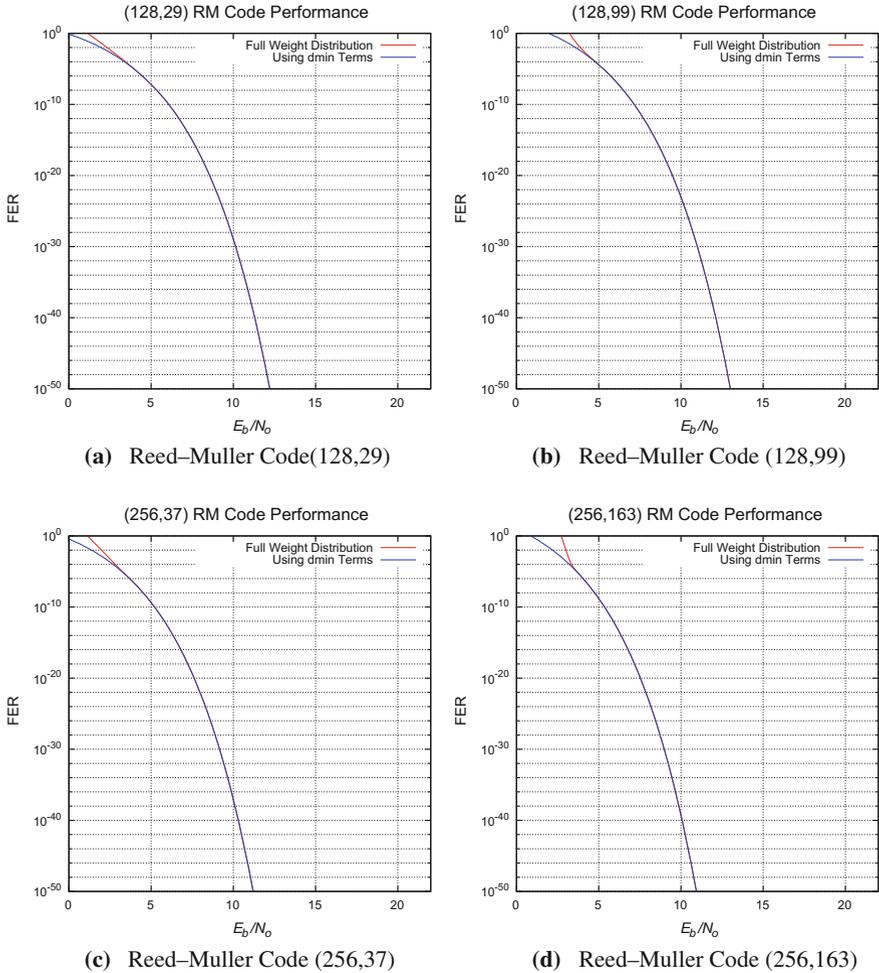
Fig. 2.5 Reed–Muller code lower and upper union bound performance, frame error rate (FER) against $\frac{E_b}{N_0}$

hard decision decoding is smaller than that of bounded distance decoding for all the values of $\frac{E_b}{N_0}$, the upper bound on the probability of error for maximum likelihood soft decision decoding agrees with the lower bound from certain values of $\frac{E_b}{N_0}$. This suggests that for soft decision decoding, the probability of error can be accurately determined by the lower union bound from a certain value of $\frac{E_b}{N_0}$. Computing the lower union bound from (2.10) requires only the knowledge of the minimum distance of the code d and the multiplicity of the minimum weight terms A_d. In practice, A_d is much easier to obtain than the complete weight distribution of the code.
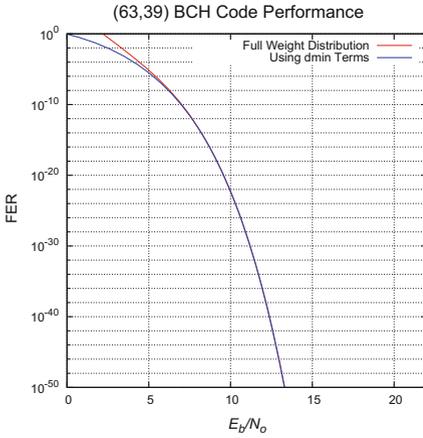
(a) BCH Code (63,39) union bounds
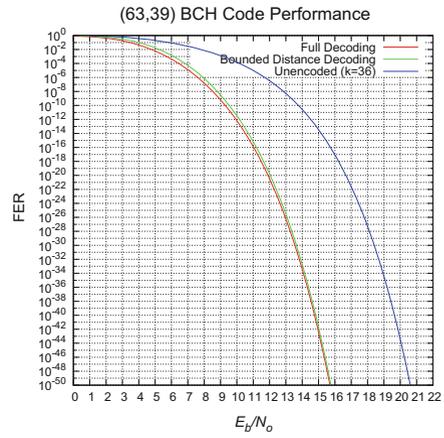
(b) BCH Code (63,39) BDD and full decoding
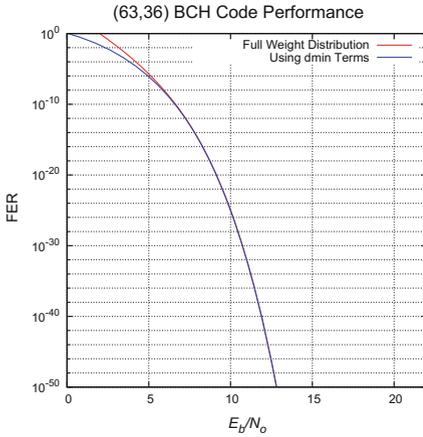
(c) BCH Code (63,36) union bounds
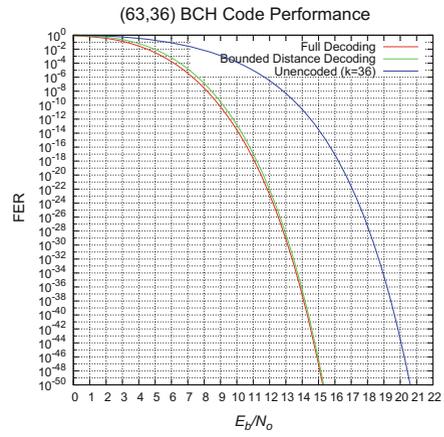
(d) BCH Code (63,36) BDD and full decoding

Fig. 2.6 BCH code: Bounded distance, full and maximum likelihood soft decoding

2.3.1 Performance Assuming a Binomial Weight Distribution

Evaluating the performance of long codes with many codewords using the union upper bound is difficult since one needs to compute the complete weight distribution of the codes. For many good linear binary codes, the weight distributions of the codes closely approximates to a binomial distribution. Computing the weight distribution of a binary code is known to be NP-complete [3]. Let $\left(\frac{E_b}{N_0}\right)_\delta$ be defined as,

(a) (127,30,37) binomial and actual weight distributions

(b) (127,120,3) binomial and actual weight distributions

(c) (70,30,16) binomial and actual weight distributions

(d) (255,235,6) binomial and actual weight distributions
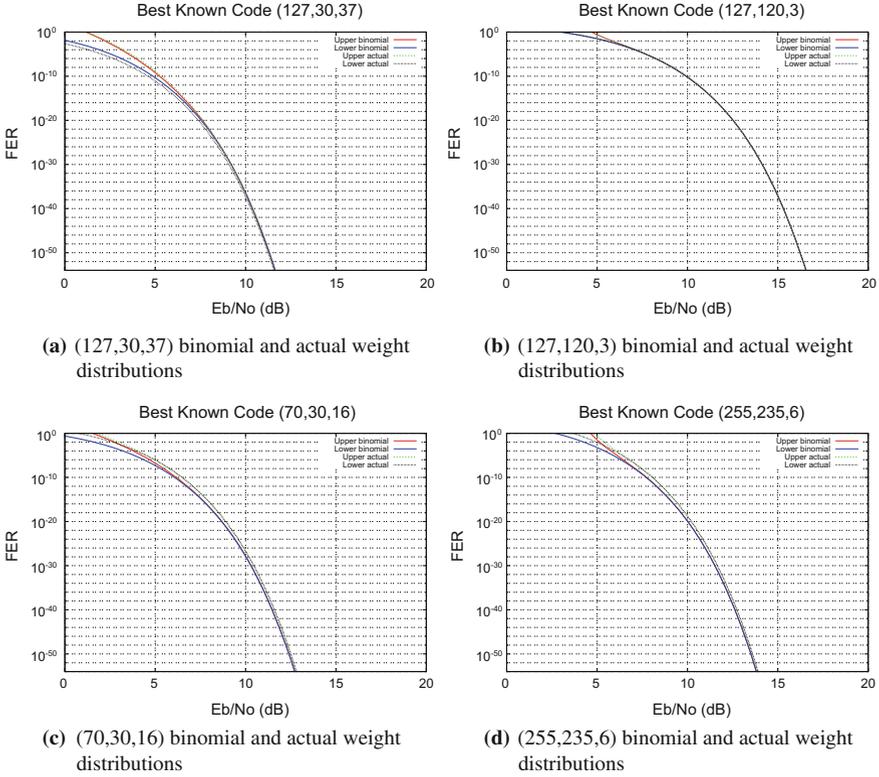
Fig. 2.7 Union bounds using binomial and actual weight distributions (WD) for best known codes

$$\frac{1}{2} A_d \operatorname{erfc}\left(\sqrt{\frac{E_b}{N_0}Rd}\right)\Bigg|_{\frac{E_b}{N_0}=\left(\frac{E_b}{N_0}\right)_\delta} \approx \frac{1}{2}\sum_{j=d}^{n} A_j \operatorname{erfc}\left(\sqrt{\frac{E_b}{N_0}Rj}\right)\Bigg|_{\frac{E_b}{N_0}=\left(\frac{E_b}{N_0}\right)_\delta}. \quad (2.12)$$

Hence, $\left(\frac{E_b}{N_0}\right)_\delta$ is the SNR per bit at which the difference between upper and lower union bound for the code is very small. It is worth noting that equality is only possible when $\frac{E_b}{N_0}$ approaches infinity in (2.12) since $\lim_{x\to\infty}\operatorname{erfc}(x) = 0$. To find $\left(\frac{E_b}{N_0}\right)_\delta$ for a binary code (n, k, d) we simply assume a binomial weight distribution for the code so that,

$$A_i = \frac{2^k}{2^n}\binom{n}{i} \quad (2.13)$$
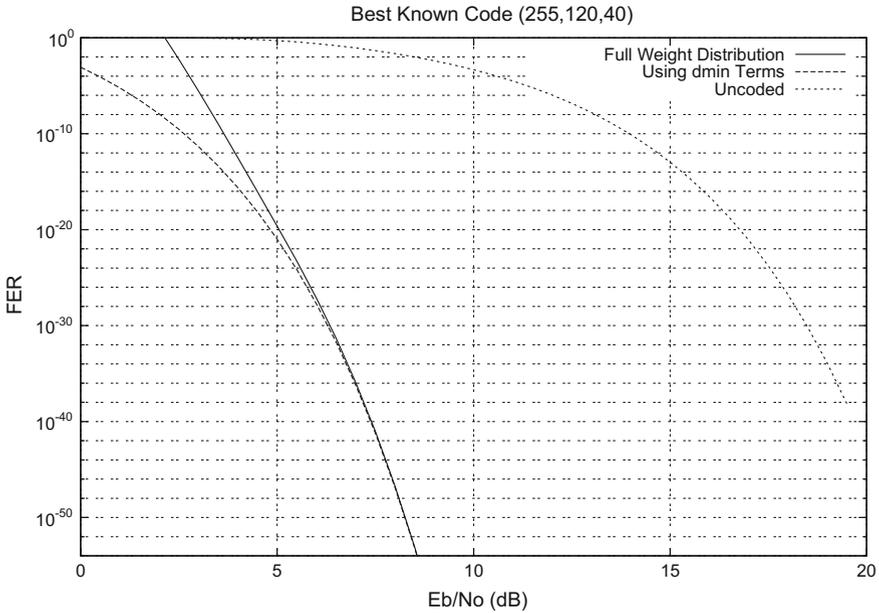
Fig. 2.8 Union bounds using binomial and actual weight distributions (WD) for the (255, 120, 40) best known code

and compute an $\frac{E_b}{N_0}$ value that satisfies (2.12). It must be noted that $\left(\frac{E_b}{N_0}\right)_\delta$ obtained using this approach is only an estimate. The accuracy of $\left(\frac{E_b}{N_0}\right)_\delta$ depends on how closely the weight distribution of the code approximates to a binomial and how small the difference between the upper and lower union bounds $P_{\text{upper}} - P_{\text{lower}}$ is. Consider Fig. 2.7 that show the upper and lower union bounds using binomial weight distributions and the actual weight distributions of the codes. From Fig. 2.7a, it can be seen that for the low rate code (127, 30, 37) the performance of the code using the binomial approximation of the weight distribution does not agree with the performance using the actual weight distribution at low values of $\frac{E_b}{N_0}$. Interestingly Fig. 2.7b–d show that as the rate of the codes increases the actual weight distribution of the codes approximates to a binomial. The difference in the performance of the codes using the binomial approximation and actual weight distribution decreases as $\frac{E_b}{N_0}$ increases. Figure 2.8 shows the performance of the (255, 120, 40) using a binomial weight distribution. An estimate for $\left(\frac{E_b}{N_0}\right)_\delta$ from the figure is 5.2 dB. Thus for $\frac{E_b}{N_0} \geq 5.2$ dB, we can estimate the performance of the (255, 120, 40) code under maximum likelihood soft decision decoding in the AWGN channel using the lower union bound.
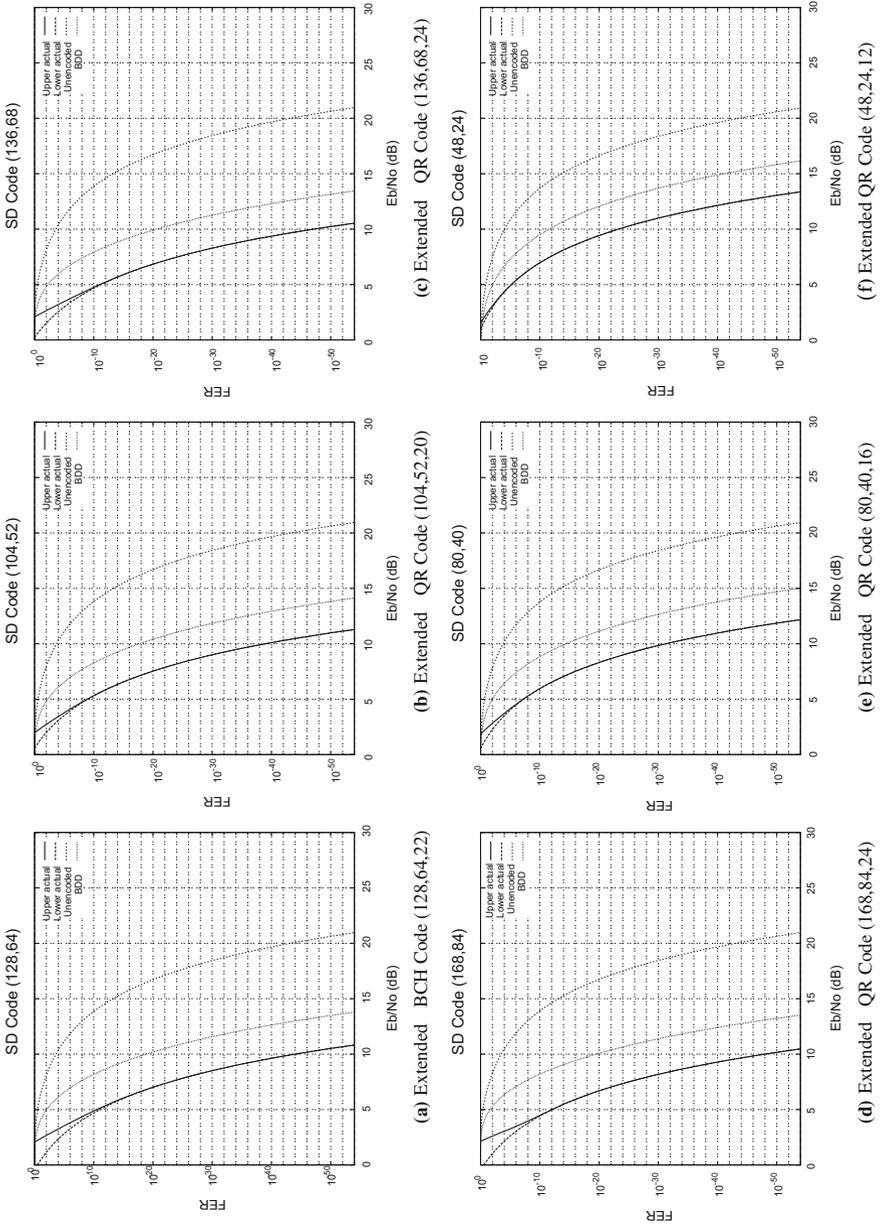
Fig. 2.9 Performance of self-dual codes

2.3.2 Performance of Self-dual Codes

A self-dual code $\mathscr{C}$ has the property that it is its own dual such that,

$$\mathscr{C} = \mathscr{C}^{\perp}.$$

Self-dual codes are always half rate with parameters $(n, \frac{1}{2}n, d)$. These codes are known to meet the Gilbert–Varshamov bound and some of the best known codes are self-dual codes. Self-dual codes form a subclass of formally self-dual codes which have the property that,

$$W(\mathscr{C}) = W(\mathscr{C}^{\perp}).$$

where $W(\mathscr{C})$ means the weight distribution of $\mathscr{C}$. The weight distribution of certain types of formally self-dual codes can be computed without enumerating all the codewords of the code. For this reason, these codes can readily be used for analytical purposes. The fact that self-dual codes have the same code rate and good properties makes them ideal for performance evaluation of codes of varying length. Consider Fig. 2.9 which shows the performance of binary self-dual (and formally self-dual) codes of different lengths using the upper and lower union bounds with actual weight distributions, bounded distance decoding and unencoded transmission. Figure 2.10
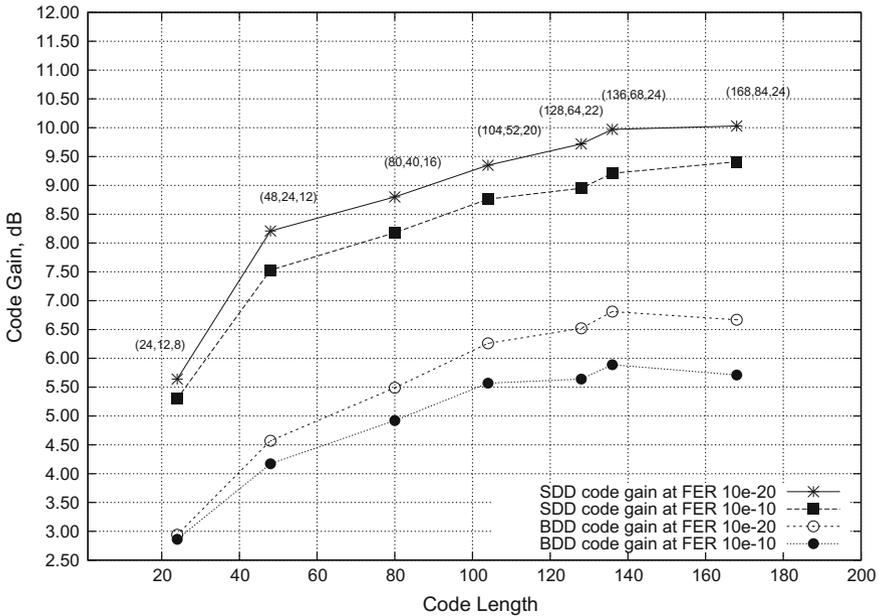
Fig. 2.10 Coding gain against code length for self-dual codes at FER 10^{-10} and 10^{-20}

shows the coding gain of the self-dual codes at frame error rates (FER) 10^{-10} and 10^{-20} for soft decision decoding (SDD) and bounded distance decoding (BDD). The coding gain represents the difference in dB between the SDD/BDD performance and unencoded transmission. The coding gain is a measure of the power saving obtainable from a coded system relative to an unencoded system in dB at a certain probability of error. The SDD performance of codes with length 168, 136 and 128 at FER 10^{-10} are obtained from the union upper bound because the upper and lower bound do not agree at this FER. Thus, the coding gain for these cases is a lower bound. It is instructive to note that the difference between the coding gain for SDD and BDD at the two values of FER increases as the length of the code increases. At FER of 10^{-20} SDD gives 3.36 dB coding gain over BDD for the code of length 168 and 2.70 dB for the code of length 24. At a FER of 10^{-10}, SDD gives 3.70 dB coding gain over BDD for the code of length 168 and 2.44 dB for the code of length 24.

2.4   Summary

In this chapter, we discussed the performance of codes under hard and soft decision decoding. For hard decision decoding, the performance of codes in the binary symmetric channel was discussed and numerically evaluated results for the bounded distance decoder compared to the full decoder were presented for a range of codes whose coset leader weight distribution is known. It was shown that as the SNR per information bit increases there is still an observable difference between bounded distance and full decoders. A lower and upper bound for decoding in the BSC was also given for cases where the covering radius of the code is known. For soft decision decoding, the performance of a wide range of specific codes was evaluated numerically using the union bounds. The upper and lower union bounds were shown to converge for all codes as the SNR per information bit increases. It was apparent that for surprisingly low values of $\frac{E_b}{N_0}$ the performance of a linear code can be predicted by only using knowledge of the multiplicity of codewords of minimum weight. It was also shown for those codes whose weight distribution is difficult to compute, a binomial weight distribution can be used instead.

References

1. Moon, T.K.: Error Correction Coding: Mathematical Methods and Algorithms. Wiley, New Jersey (2005)
2. Proakis, J.: Digital Communications, 4th edn. McGraw-Hill, New York (2001)
3. Vardy, A.: The intractability of computing the minimum distance of a code. IEEE Trans. Inf. Theory **43**, 1759–1766 (1997)