CHAPTER 3
LOW-DENSITY BURST ERROR CORRECTING AND LOCATING CODES

The contents of this chapter include matter from my following research papers:


CHAPTER 3

LOW-DENSITY BURST ERROR CORRECTING AND LOCATING CODES

As pointed out in the preceding chapter, in some communication systems errors occur predominantly in bursts. Codes developed to correct such errors are called burst-error-correcting codes.

Lightning or other disturbances which introduce burst errors usually operate in a way such that, over a given length, some digits are received correctly while others are corrupted. In such situations, provision for correcting all digits in a burst of specified length is not required. If the usual burst correcting codes are employed in such situations, the efficiency of transmission is effected adversely. Such situations therefore demand the development of codes which correct those errors that are bursts of a specified length, say b, with weight w or less (w ≤ b). Alternatively, the detection, correction and location of w or less random errors in bursts of length b is needed. The development of such codes can economize in parity-check digits required, suitably reducing the redundancy of the code. Such codes have been termed as low-density burst correcting codes and were initiated by Wyner (1963).

A study of burst-error correcting and locating codes keeping in view the weight constraint over them has been
made in this chapter. This generalizes not only the study of burst error control but significantly generalizes the concept of random error correction, since it is considered over burst length rather than over word length.

In the first section of this chapter, we shall deal with linear codes capable of correcting low-density burst errors. Lower and upper bounds over the number of parity-check digits for such codes have been obtained. In carrying out this study, we shall confine to the definition 2.3 of a burst proposed in the preceding chapter viz. 'A burst of length $b(\text{fixed})$ is an $n$-tuple whose only nonzero components are confined to some $b$ consecutive digits, the first of which is nonzero and the number of its starting positions is the first $(n-b+1)$ positions'.

The second section of this chapter presents a study of burst error locating codes, keeping in view the weight constraint over the burst length. Lower and upper bounds on the necessary and sufficient number of parity-check digits required for the existence of such a code have been obtained. The study of error-locating codes has been carried out with respect to the usual definition of open-loop bursts.

3.1. Bounds for Codes Correcting Low-Density Burst Errors

In this section, we first give a lower bound on the necessary number of check digits and then an upper bound on
the sufficient number of check digits required for a code capable of correcting all bursts of length \( b(\text{fixed}) \) with weight \( w \) or less \((w \leq b)\). Before coming to the main result, we give a lemma.

**Lemma 3.1.** The total number of bursts of length \( b(\text{fixed}) \) \((b \geq 1)\) with weight \( w \) or less is

\[
(n-b+1)(q-1) \left[ \begin{array}{c} 1^+ (q-1) \\ (b-1, w-1) \end{array} \right].
\]

\[(3.1)\]

where \([1+x]^{m,r}\) denotes the incomplete binomial expansion of \((1+x)^m\) up to the term \(x^r\) in ascending powers of \(x\).

**Proof.** The lemma follows immediately since the number of bursts of length \( b(\text{fixed}) \) with weight \( i \) is

\[
\binom{b-1}{i-1} (q-1)^i (n-b+1).
\]

\[Q.E.D.\]

**Theorem 3.1.** The number of parity check symbols in an \((n,k)\) linear code that corrects all bursts of length \( b(\text{fixed}) \) with weight \( w \) or less \((w \leq b)\) is at least

\[
\log_q \left[ 1^+ (q-1)(n-b+1) \left[ \begin{array}{c} 1^+ (q-1) \\ (b-1, w-1) \end{array} \right] \right].
\]

\[(3.2)\]

where the notation \([1+x]^{m,r}\) has the meaning stated in Lemma 3.1.
Proof. Since the code is capable of correcting all errors which are bursts of length $b$ (fixed) with weight $w$ or less, all such patterns should be in different cosets of the standard array. Using the lemma, the total number of bursts of length $b$ (fixed) with weight $w$ or less, including the pattern of all zeros, is

$$1 + (q-1)(n-b+1) \lceil \frac{b}{w} \rceil^{(b-1,w-1)}.$$

The theorem now follows from the fact that there must be at least this number of cosets in the standard array. Q.E.D.

Incidentally, it can be shown that the result applies to non-linear codes also.

For $w = b$, the lower bound on the number of parity checks becomes

$$\log_q \left[ 1 + (q-1)(n-b+1)q^{b-1} \right]$$

which coincides with the result obtained in Theorem 2.1 of Chapter 2 (also refer to Dass (1932)).

We now derive an upper bound on the number of parity check digits that assures the existence of such a low-density burst error correcting code. The proof involves a suitable modification of the technique used by Sacks (1958) in establishing the well-known Varshamov-Gilbert bound.
Theorem 3.2  Given positive integers \( w \) and \( b \) such that \( w \leq b \), there exists an \((n,k)\) linear code that corrects all bursts of length \( b(fixed) \) with weight \( w \) or less satisfying the inequality

\[
q^{n-k} > [1+(q-1)^{(b-1)(w-l)}][1+(q-1)(n-2b+1)][1+(q-1)^{(b-1)(w-l)}]
\]

\[
+ \sum_{i=w}^{2w-1} (b-1) (q-1)^i
\]

\[
+ \sum_{k=1}^{b-1} \sum_{r_1, r_2, r_3} (b-k) (r_1^{e-1}) (r_2^{b-k-1}) (r_3^{e-1}) (q-1)\]

\[
1 \leq r_1 \leq w - 1
\]
\[
0 \leq r_2 \leq 2w - 3
\]
\[
0 \leq r_3 \leq w - 1
\]
\[
r_2 + r_3 \geq w - 1
\]
\[
r_1 + r_2 + r_3 \leq 2w - 2,
\]

where \([b\times x]^{(m,r)}\) has the meaning stated in Lemma 3.1.

Proof. The existence of such a code will be shown by constructing an appropriate \((n-k) \times n\) parity-check matrix \(H\). In order to do so, we first construct a matrix \(H'\), from which we shall obtain the requisite parity-check matrix \(H\) by reversing all the columns of \(H'\).
A nonzero \((n-k)\)-tuple is chosen as the first column of \(H'\). Subsequent columns are added such that after having selected \(j-1\) columns \(h_1, h_2, \ldots, h_{j-1}\), a column \(h_j\) is added provided that it is not a linear combination of any \(w-1\) or fewer columns among the immediately preceding \(b-1\) columns together with any \(w\) or fewer columns among any \(b\) consecutive columns. (Here \(b\) consecutive columns does not include less than \(b\) columns). In other words, \(h_j\) is added provided that

\[
h_j \neq (a_{i_1} h_{i_1} + a_{i_2} h_{i_2} + \ldots + a_{i_{w-1}} h_{i_{w-1}}) \\
\quad + (b_{j_1} h_{j_1} + b_{j_2} h_{j_2} + \ldots + b_{j_w} h_{j_w}),
\]  

\[ (3.4) \]

where the \(h_i\) are any \(w-1\) columns among \(h_{j-b+1}, h_{j-b+2}, \ldots, h_{j-1}\), and the \(h_j\) are any \(w\) columns from a set of \(b\) consecutive columns among all the \(j-1\) columns such that either all the \(b_j\) are zero or if \(b_t\) is the last nonzero coefficient then

\[ b \leq t \leq j-1. \]

(The restriction \(t \leq j-1\) is obviously satisfied since the selection of the columns is out of all the \(j-1\) previously chosen columns).

To obtain the number of all possible distinct linear combinations, we analyze three different cases.
Case I. When the \( h_j \) are taken from the first \( j-b \) columns.

In this case, the number of ways in which the coefficients \( a_1 \) can be selected is

\[
[1 + (q-1)(b-1,w-1)]^{(b-1,w-1)}, \quad (3.5)
\]

whereas the number of ways in which the \( b_j \), which form a burst of length \( b \) (fixed) with weight \( w \) or less in a vector of length \( j-b \), can be selected is (refer Theorem 3.1 of this chapter)

\[
1 + (q-1)(j-2b+1)[1+(q-1)]^{(b-1,w-1)}. \quad (3.6)
\]

Therefore, the total number of choices of coefficients in this case is

\[
[1+(q-1)]^{(b-1,w-1)}[1+(q-1)(j-2b+1)[1+(q-1)]^{(b-1,w-1)}]^{(b-1,w-1)} \quad \ldots \quad (3.7)
\]

Case II. When the \( h_j \) are taken from the immediately preceding \( b-1 \) columns \( h_j-b+1, h_j-b+2, \ldots, h_j-1 \).

In this case, we have to select the coefficients \( a_1 \) and \( b_j \), which are \( 2w-1 \) or less in number, from amongst \( b-1 \) components. Since \( w-1 \) or less coefficients have already been accounted for in (3.5), the additional number of ways to choose \( a_1 \) and \( b_j \) is
\[
\sum_{i=w}^{2w-1} \binom{b-1}{1}(q-1)^i.
\]

(3.8)

**Case III.** When the \( h_j \) are selected from \( h_{j-2b+2}, h_{j-2b+3}, \ldots, h_{j-1} \) such that all the \( h_j \) are neither taken from \( h_{j-2b+2}, h_{j-2b+3}, \ldots, h_{j-b} \) nor from \( h_{j-b+1}, h_{j-b+2}, \ldots, h_{j-1} \).

In this case, let us suppose that the burst starts from the \((j-2b+k+1)\)-th component which may obviously continue up to the \((j-b+k)\)-th component. Our object is to select \( w-1 \) or less nonzero components among \((j-2b+k+1, j-2b+k+2, \ldots, j-b+k-1)\)-th position together with \( w-1 \) or less nonzero components among \((j-b+1, j-b+2, \ldots, j-1)\)-th position.

\[
\begin{array}{cccccccc}
j-2b+2 & j-2b+k+1 & j-b & j-b+1 & j-b+k & j-1 \\
\end{array}
\]

\[\leftarrow b \rightarrow\]

In order to do so, let us choose \( r_1 \) components from the \((j-2b+k+1, \ldots, j-b)\)-th position, \( r_2 \) components from the \((j-b+1, \ldots, j-b+k-1)\)-th position and \( r_3 \) components from the \((j-b+k+1, \ldots, j-1)\)-th position, where

\[
\begin{align*}
1 \leq r_1 & \leq w-1 \\
0 \leq r_2 & \leq 2w-3 \\
0 \leq r_3 & \leq w-1.
\end{align*}
\]

(3.9)
Keeping in view the situations considered in cases 1 and 2, \( r_1, r_2, r_3 \) should be such that

\[
\begin{align*}
\left\{ \begin{array}{c}
{r_2}^+ r_3 & \geq w-1 \\
{r_1}^+ r_2^+ r_3 & \leq 2w - 2
\end{array} \right. \tag{3.10}
\end{align*}
\]

Such a selection of coefficients gives us

\[
\sum_{r_1, r_2, r_3} (b-k)(k-1)(b-k-1)(q-1)^{r_1^+r_2^+r_3}
\]

possible linear combinations where \( r_1, r_2, r_3 \) each satisfy the constraints stated in (3.9) and (3.10).

Also, the \((j-b+k)\)-th component can be selected in \((q-1)\) ways.

Then, in this case, the total number of choices of coefficients turns out to be

\[
\sum_{k=1}^{b-1} \sum_{r_1, r_2, r_3} (b-k)(k-1)(b-k-1)(q-1)^{r_1^+r_2^+r_3+1}. \tag{3.11}
\]

Thus, the total possible number of distinct combinations which \( h_j \) cannot equal is

\[
\left[1+(q-1)\right]^{(b-1,w-1)} \left[1+(q-1)(j-2b+1)[1+(q-1)]^{(b-1,w-1)}\right]
\]

\[
+ \sum_{i=w}^{2w-1} (b-1)(q-1)^i
\]

\[
+ \sum_{k=1}^{b-1} \sum_{r_1, r_2, r_3} (b-k)(k-1)(b-k-1)(q-1)^{r_1^+r_2^+r_3+1}. \tag{3.12}
\]
Therefore, a column $h_j$ can be added to $H'$ provided that all the $(n-k)$-tuples are not exhausted by the above forbidden combinations.

At worst, all these combinations might yield a distinct sum. Therefore, $h_j$ can always be added provided that

$$d^{n-k} > [1+(q-1)]^{(b-1,w-1)} + [1+(q-1)(j-2b+1)][1+(q-1)]^{(b-1,w-1)}$$

$$+ \sum_{i=w}^{2w-1} \binom{b-1}{i}(q-1)^i$$

$$+ \sum_{k=1}^{b-1} \sum_{r_1, r_2, r_3} \binom{b-k}{k-1} \binom{b-k-1}{r_1} \binom{b-k-1}{r_2} \binom{b-k-1}{r_3} (q-1)^{r_1+r_2+r_3+1} \cdots$$

But for an $(n,k)$ desired code to exist, the inequality (3.13) should hold for $j = n$ so that it is possible to add up to the $n$-th column to form an $(n-k) \times n$ matrix. The required parity-check matrix $H = [H_1 H_2 \cdots H_n]$, $H_i$ denoting the $i$-th column, is then the matrix obtained from $H'$ by reversing its columns altogether, i.e. $h_i \rightarrow H_{n-i+1}$ ($h_n \rightarrow H_1$, $h_{n-1} \rightarrow H_2$, ..., $h_1 \rightarrow H_n$).

This leads to the sufficient condition stated in (3.3).

Q.E.D.

The result just obtained holds for $w \leq b$. If we take $w = b$, the weight consideration over the burst becomes
redundant. The expression in (3.8) vanishes, while the situation giving rise to the expression (3.11) does not arise. The bound then reduces to

\[ q^{n-k} > [1 + (q-1)]^{(b-1, b-1)} \]

\[ \cdot [1 + (q-1)(n-2b+1)[1 + (q-1)]^{(b-1, b-1)} \]

i.e., \[ q^{n-k} > q^{b-1} [1 + (q-1)(n-2b+1)q^{b-1}] \]

which coincides with the result proved in Theorem 2.2 of chapter 2 (also refer to Dass (1980)).

**Alternate Form 3.1.** If \( n \) is the largest value of \( j \) satisfying the inequality (3.13), then for \( j = n + 1 \), the inequality in (3.3) gets reversed and we get

\[ q^{n-k} \leq [1 + (q-1)]^{(b-1, w-1)} [1 + (q-1)(n-2b+2)[1 + (q-1)]^{(b-1, w-1)}] \]

\[ + \sum_{i=w}^{2w-1} \binom{b-1}{i}(q-1)^i \]

\[ + \sum_{k=1}^{b-1} \frac{(b-k)(k-1)}{r_1, r_2, r_3} \frac{(b-k-1)}{r_1, r_2, r_3} \frac{(q-1)}{r_1, r_2, r_3} r_1 + r_2 + r_3 + 1 \]

\[ \ldots \quad (3.14) \]

**Alternate Form 3.2.** If \( B \) is the largest value of \( b \) satisfying the inequality (3.3), then for \( b = B + 1 \) also,
the inequality in (3.3) gets reversed, and we get

\[ q^{n-k} \leq \left[ 1 + (q-1) \right]^{(B_w-1)} \left[ 1 + (q-1)(n-2B-1) [1 + (q-1)]^{(B, w-1)} \right] \]

\[ + \sum_{i=w}^{2w-1} \binom{B}{i} (q-1)^i \]

\[ + \sum_{k=1}^{B} \sum_{r_1, r_2, r_3} (B-k+1)(k-1)(B-k)(q-1)^{r_1+r_2+r_3+1} \]

\[ \ldots \quad (3.15) \]

In Table 3.1 for \( b = 3 \) and \( w = 2 \) we give, for specific values of \( n \) upto 20, the maximum number of information digits, \( k \), possible for the class of binary codes of the type stated in Theorem 3.2, correcting bursts of length \( b(\text{fixed}) \) with weight \( w \) or less. For the purpose of comparison, we also give, \( k^* \), the maximum number of information digits possible for binary codes that correct all bursts of length \( b \) or less with weight \( w \) or less with respect to the definition of an open-loop burst. The result for open-loop low-density bursts over GF(q) proved by Dass (1975) is as follows:

**Result.** Given positive integers \( w \) and \( b \) such that \( w \leq b \), there exists an \((n,k)\) linear code that corrects all open-loop bursts of length \( b \) or less with weight \( w \) or less satisfying the inequality.
\[ q^{n-k} > [1+(q-1)]^{(b-1, w-1)} \left[ q^{w-1} [(q-1)(n-b-w+1) + 1] \right. \\
\left. + (q-1)^2 \sum_{i=w+1}^{b} (n-b-i+1) [1+(q-1)]^{i-2, w-2} \right] \\
+ \sum_{i=w}^{2w-1} (\binom{b-1}{i} (q-1)^i) \\
+ \sum_{k=1}^{b-1} \sum_{r_1, r_2, r_3} (\frac{b-k}{r_1}) (\frac{k}{r_2}) (\frac{b-k-1}{q-1}) (\frac{r_1+r_2+r_3+1}{r_3}) \\
\text{where } [1+x]^{m, r} \text{ has the meaning stated in Lemma 3.1.} \\
\]
We have listed only those values of \( n \) up to 20 for which the values of \( k \) and \( k^* \) differ. For \( n = 12, 19, 20 \), the values of \( k \) and \( k^* \) are equal.

We conclude this section with an example.

Example 3.1. Consider the following 5 x 8 matrix over \( GF(2) \).

\[
\begin{bmatrix}
0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\
0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \\
0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 \\
0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 \\
1 & 0 & 0 & 0 & 0 & 1 & 0 & 0
\end{bmatrix}
\]

This matrix has been constructed by the synthesis procedure, outlined in the proof of Theorem 3.2 by taking \( b = 3 \) and \( w = 2 \). Considered as a parity check matrix, this matrix gives rise to a \((8,3)\) binary code. It can be seen from Table 3.2 that the syndromes of bursts of length 3(fixed) with weight 2 or less are distinct, showing thereby that the code that is the null space of the matrix given in the example corrects all bursts of length 3(fixed) with weight 2 or less. It should be noted that this code does not correct all bursts of length 3(fixed) with weight 3, e.g. \((11100000)\), as its syndrome is the same as that of \((00000100)\).
Table 3.2

<table>
<thead>
<tr>
<th>Error Pattern</th>
<th>Syndrome</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 0 1 0 0 0 0 0</td>
<td>0 0 1 0 1</td>
</tr>
<tr>
<td>1 1 0 0 0 0 0 0</td>
<td>0 0 0 1 1</td>
</tr>
<tr>
<td>1 0 0 0 0 0 0 0</td>
<td>0 0 0 0 1</td>
</tr>
<tr>
<td>0 1 0 1 0 0 0 0</td>
<td>0 1 0 1 0</td>
</tr>
<tr>
<td>0 1 1 0 0 0 0 0</td>
<td>0 0 1 1 0</td>
</tr>
<tr>
<td>0 1 0 0 0 0 0 0</td>
<td>0 0 0 1 0</td>
</tr>
<tr>
<td>0 0 1 0 1 0 0 0</td>
<td>1 0 1 0 0</td>
</tr>
<tr>
<td>0 0 1 1 0 0 0 0</td>
<td>0 1 1 0 0</td>
</tr>
<tr>
<td>0 0 1 0 0 0 0 0</td>
<td>0 0 1 0 0</td>
</tr>
<tr>
<td>0 0 0 1 0 1 0 0</td>
<td>0 1 1 1 1</td>
</tr>
<tr>
<td>0 0 0 1 1 0 0 0</td>
<td>1 1 0 0 0</td>
</tr>
<tr>
<td>0 0 0 1 0 0 0 0</td>
<td>0 1 0 0 0</td>
</tr>
<tr>
<td>0 0 0 0 1 0 1 0</td>
<td>1 1 1 1 0</td>
</tr>
<tr>
<td>0 0 0 0 1 1 0 0</td>
<td>1 0 1 1 1</td>
</tr>
<tr>
<td>0 0 0 0 1 0 0 0</td>
<td>1 0 0 0 0</td>
</tr>
<tr>
<td>0 0 0 0 0 1 0 1</td>
<td>1 1 0 1 1</td>
</tr>
<tr>
<td>0 0 0 0 0 1 1 0</td>
<td>0 1 0 0 1</td>
</tr>
<tr>
<td>0 0 0 0 0 1 0 0</td>
<td>0 0 1 1 1</td>
</tr>
</tbody>
</table>

Discussion. The results presented in this section can be extended to multiple-burst correction, see Bridwell and Wolf (1970). There is an associated problem for correcting bursts of weight \( w \) or more. In the case of open-loop burst error correction, this problem leads to two interesting classes of codes:

(i) 'Binary-adjacent-error-correcting codes' that are in a sense 'perfect', see Sharma and Dass (1977),
(ii) 'Binary anti-perfect codes' that are also in a sense 'perfect', see Sharma, Dass and Gupta (1977).

It is worth examining the existence of some such a similar class of codes for the type of bursts considered in this section.

3.2. Bounds for Codes Locating Low-Density Burst Errors

A study of error-locating codes in which errors occur in the form of bursts was made in section 2.2 of the preceding chapter. As pointed out in the beginning of this chapter, a plausible model for a digital communication channel is one in which disturbances occur in bursts, and during the duration of these disturbances the error probability is not large. Mathematical models for similar channels have been proposed by Gilbert (1960) and Elspas (1961). This section presents a study of error-locating codes, keeping in view the weight constraint over the burst length. Lower and upper bounds on the necessary and sufficient number of parity check digits required for the existence of such a code have been obtained.

In the following text, a burst of length $b$ will be regarded as an open-loop burst of length $b$, e.g. an $n$-tuple all of whose nonzero components are confined to $b$ consecutive positions, the first and the last of which are nonzero.
The block of \( n \) digits, consisting of \( r \) check digits, and \( k = n-r \) information digits, is subdivided into \( s \) mutually exclusive sub-blocks. Each sub-block contains \( t = \frac{n}{s} \) digits.

An error-locating code capable of detecting and locating a single sub-block containing errors that are burst of length \( b \) or less with weight \( w \) or less must satisfy the following conditions:

(a) The syndrome resulting from the occurrence of an error which is a burst of length \( b \) or less with weight \( w \) or less within any one sub-block must be distinct from the all-zeros syndrome.

(b) The syndrome resulting from the occurrence of a burst error of length \( b \) or less with weight \( w \) or less within a single sub-block must be distinct from the syndrome resulting from any other burst of length \( b \) or less with weight \( w \) or less within some other sub-block.

In the following, we shall derive two results. The first result gives a lower bound on the number of check digits required for the existence of a linear code over \( \text{GF}(q) \) capable of detecting and locating a single sub-block containing errors that are bursts of length \( b \) or less with weight \( w \) or less. In the second result, we derive an upper bound on the number of check digits, which assures the existence of such a code.
Theorem 3.3. The number of parity-check digits \( r \) is an \((n,k)\) linear code over \( \text{GF}(q) \), subdivided into \( s \) sub-blocks of length \( t \) each, that locates a single corrupted sub-block containing errors that are bursts of length \( b \) or less, with weight \( w \) or less \((w \leq b)\) is at least

\[
\log_q [s(q^w - 1)]. \quad (3.16)
\]

**Proof.** The maximum number of distinct syndromes available using \( r \) check digits is \( q^r \). The proof proceeds by first counting the number of syndromes that are required to be distinct by conditions (a) and (b) and then sets this number as less than or equal to \( q^r \).

Let us fix some \( w \) positions out of the first \( b \) components of every sub-block, each of length \( t \). Since the code detects all bursts of length \( b \) or less, with weight \( w \) or less, lying in any sub-block, any two vectors that are zero except in the fixed \( w \) positions out of the first \( b \) positions of a particular sub-block cannot lie in different cosets of the standard array; otherwise their difference would be a burst of length \( b \) or less with weight \( w \) or less, resulting in the zero syndrome (in violation of condition (a)). Further, since the code locates a single sub-block containing errors that are bursts of length \( b \) or less, with weight \( w \) or less, syndromes produced by burst errors of length \( b \) or less, with weight \( w \) or less,
in different sub-blocks must be distinct by condition (b). This will certainly be true for syndromes of errors of weight \( w \) or less (nonzero positions restricted to the fixed \( w \) components) that are confined to the first \( b \) positions of any sub-block. Thus, the syndromes of errors that are confined to the fixed \( w \) positions out of the first \( b \) positions of any sub-block, whether in the same or in different sub-blocks, must be distinct. Since there are \( q^w - 1 \) possible errors in the first \( b \) positions of any sub-block, and there are \( s \) sub-blocks in all, there must be

\[
1 + s(q^w - 1)
\]

distinct syndromes, counting the all-zero syndrome.

Thus, we must have

\[
q^r \geq 1 + s(q^w - 1)
\]

from which the result follows by taking logarithms. \( \text{Q.E.D.} \)

The result just obtained holds for \( w \leq b \). If we take \( w = b \), the weight consideration over the burst becomes redundant and the bound then reduces to

\[
\log_q [1 + s(q^b - 1)]
\]

which coincides with the result proved in Theorem 2.3 of chapter 2 (also refer to Nass (1982)).
Discussion.

(i) It is worth noticing that the bound obtained in the preceding theorem is independent of \( b \) and \( t \). Thus, the result obtained in expression (3.16) remains valid for all values of \( b \) and \( t \) so long as \( w \leq b \leq t \) and \( n = st \).

(ii) It is not necessary to fix the same \( w \) positions out of the first \( b \) positions of every sub-block. Fixing up different \( w \) positions out of the first \( b \) components of various sub-blocks yields the same result.

An upper bound on the number \( r \) of check digits required is given by the theorem 3.4. The proof involves relative modifications of the procedure used to establish the Varshamov-Gilbert-Sacks bound (see Peterson and Weldon Jr. (1972), c.f. theorem 4.7).

Theorem 3.4. A code capable of detecting burst errors of length \( b \) or less, with weight \( w \) or less (\( w \leq b \)), occuring within a single sub-block, and of locating that sub-block, can always be constructed using \( r \) check digits where \( r \) is the smallest integer satisfying the inequality

\[
q^r > \left[ \sum_{i=0}^{w-1} \left( \begin{array}{c} b-1 \\ i \\ \end{array} \right) (q-1)^i \right] \left[ 1 + (s-1) \right] \left[ q^{w-1} \left( \begin{array}{c} (q-1)(t-w+1)+1 \\ 1 \\ \end{array} \right) \right] \\
+ (q-1)^2 \sum_{j=w+1}^{b} \sum_{i=0}^{w-2} \left( \begin{array}{c} b \\ \right) \left( \begin{array}{c} w-2 \\ \right) \left( \begin{array}{c} j-2 \\ \end{array} \right) (q-1)^{i-1} \right] (3.17)
\]
Proof. Let the required code be the null space of a matrix $H$. Suppose that we have added to $H$ the first $(s-1)t$ columns corresponding to the first $(s-1)$ sub-blocks and the first $(j-1)$ columns $h_j, h_{j+1}, \ldots, h_{j+t-1}$ of the $s$-th sub-block.

For the detection, the syndrome resulting from the occurrence of $w$ or fewer positions out of $b$ or fewer consecutive positions within any sub-block must be different from the all zero $r$-tuple. So, the $j$-th column $h_j$ can be added to the $s$-th sub-block, provided that

(i) $h_j$ is not a linear combination of any $(w-1)$ or fewer columns out of the immediately preceding $(b-1)$ columns of the $s$-th sub-block.

For locating the corrupted sub-block, the syndrome of any burst of length $b$ or less with weight $w$ or less within a sub-block must be different from the syndrome resulting from any other burst of length $b$ or less, with weight $w$ or less, within any other sub-block. So, the $j$-th column $h_j$ to be added to the $s$-th sub-block should be such that

(ii) $h_j$ is not a linear combination of $(w-1)$ or fewer columns out of the immediately preceding $(b-1)$ columns of the $s$-th sub-block, together with a linear combination of any $w$ or fewer columns out of any $b$ or fewer consecutive columns from among any one of the remaining $(s-1)$ sub-blocks.
Condition (i) gives rise to

$$\sum_{i=0}^{w-1} \ Binomial {b-1}{1} (q-1)^i$$

(3.18)

possible combinations, whereas requirement (ii) gives rise to

$$\left[ \sum_{i=0}^{w-1} \ Binomial {b-1}{1} (q-1)^i \right] (s-1) \left[ q^{w-1} [(q-1)(t-w+1)+1] \right.$$

$$+ (q-1)^2 \sum_{j=w+1}^{b} (t-j+1) \sum_{i=0}^{w-2} \ Binomial {j-2}{1} (q-1)^i -1 \right]$$

(3.19)

possible linear combinations (refer to theorem 2 of Sharma and Dass (1974)). At worst, all these linear combinations might yield a distinct sum. Thus, a column $h_j$ can be added to the $s$-th sub-block of $H$, provided that all the $r$-tuples are not exhausted by these linear combinations, i.e., for $j = t$, the $t$-th column $h_t$ can be added to the $s$-th sub-block if

$$q^r > \sum_{i=0}^{w-1} \ Binomial {b-1}{1} (q-1)^i$$

$$+ \left[ \sum_{i=0}^{w-1} \ Binomial {b-1}{1} (q-1)^i \right] (s-1) \left[ q^{w-1} [(q-1)(t-w+1)+1] \right.$$

$$+ (q-1)^2 \sum_{j=w+1}^{b} (t-j+1) \sum_{i=0}^{w-2} \ Binomial {j-2}{1} (q-1)^i -1 \right]$$

and thus we obtain the expression (3.17) by taking logarithms. Q.E.D.
If we take \( w = b \) in the preceding theorem, the weight consideration over the burst becomes redundant. The term

\[
\sum_{j=w+1}^{b} (t-j+1) \sum_{i=0}^{w-2} (j-2)(q-1)^i
\]

in expression (3.19) vanishes, and the bound in expression (3.17) then reduces to

\[
q^r > q^{b-1} \left[ 1 + (s-1) \left\{ q^{b-1} \left[ (q-1)(t-b+1) + 1 \right] - 1 \right\} \right],
\]

which has been proved independently in Theorem 2.4 of chapter 2 (also refer to Dass (1982)).

**Alternative Form 3.3.** If \( B \) is the largest value of \( b \) satisfying the inequality (3.17), then for \( b = B + 1 \) this inequality gets reversed and we get

\[
q^r \leq \left[ \sum_{i=0}^{w-1} (B+1)(q-1)^i \right] \left[ 1 + (s-1)q^{w-1} \left[ (q-1)(t-w+1) + 1 \right] \right]
\]

\[
+ (q-1)^2 \sum_{j=w+1}^{B+1} (t-j+1) \sum_{i=0}^{w-2} (j-2)(q-1)^i - 1 \right\} \right].
\]

In Table 3.3, we have listed for \( b = 3 \) and \( w = 2 \) and for specific values of \( n \), the number of parity check digits, \( r \), required for the class of binary codes of the type stated in Theorem 3.4, locating a sub-block containing
burst errors of length 3 or less with weight 2 or less. The block of n digits is considered to be subdivided into 2 mutually exclusive sub-blocks of length \( t = \frac{n}{2} \) digits. Obviously, the value of n in this case is even.

### Table 3.3

<table>
<thead>
<tr>
<th>( n ) (only even values of n to be considered)</th>
<th>( r )</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>5</td>
</tr>
<tr>
<td>10 - 14</td>
<td>6</td>
</tr>
<tr>
<td>16 - 28</td>
<td>7</td>
</tr>
<tr>
<td>30 - 56</td>
<td>8</td>
</tr>
<tr>
<td>58 - 114</td>
<td>9</td>
</tr>
<tr>
<td>116 - 228</td>
<td>10</td>
</tr>
</tbody>
</table>

We conclude this chapter with an example.

**Example 3.2.** Consider the following 5 x 8 matrix over GF(2).

\[
\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\
0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\
0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \\
0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 1
\end{bmatrix}
\]

This matrix, considered as a parity check matrix, gives rise to a (8,3) binary code. This matrix has been constructed by the synthesis procedure outlined in the proof of Theorem 3.4, by taking \( b = 3 \) and \( w = 2 \). So, the code which is the null
space of this matrix, can be used to locate a sub-block of length 4 containing burst errors of length 3 or less with weight 2 or less. It is evident from the Error Pattern-Syndrome table 3.4 that

(i) the syndrome of any burst of length 3 or less with weight 2 or less within any sub-block is nonzero,

(ii) the syndrome of a burst of length 3 or less with weight 2 or less within any sub-block is different from the syndrome of a burst of length 3 or less with weight 2 or less within the other sub-block.

<table>
<thead>
<tr>
<th>Error Pattern</th>
<th>Syndrome</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 0 0 0 0 0 0 0</td>
<td>1 0 0 0 0</td>
</tr>
<tr>
<td>1 1 0 0 0 0 0 0</td>
<td>1 1 0 0 0</td>
</tr>
<tr>
<td>1 0 1 0 0 0 0 0</td>
<td>1 0 1 0 0</td>
</tr>
<tr>
<td>0 1 0 0 0 0 0 0</td>
<td>0 1 0 0 0</td>
</tr>
<tr>
<td>0 1 1 0 0 0 0 0</td>
<td>0 1 1 0 0</td>
</tr>
<tr>
<td>0 1 0 1 0 0 0 0</td>
<td>0 1 0 1 0</td>
</tr>
<tr>
<td>0 0 1 0 0 0 0 0</td>
<td>0 0 1 0 0</td>
</tr>
<tr>
<td>0 0 1 1 0 0 0 0</td>
<td>0 0 1 1 0</td>
</tr>
<tr>
<td>0 0 0 1 0 0 0 0</td>
<td>0 0 0 1 0</td>
</tr>
<tr>
<td>0 0 0 0 1 0 0 0</td>
<td>0 0 0 0 1</td>
</tr>
<tr>
<td>0 0 0 0 1 1 0 0</td>
<td>1 0 0 1 1</td>
</tr>
<tr>
<td>0 0 0 0 1 0 1 0</td>
<td>1 1 1 0 1</td>
</tr>
<tr>
<td>0 0 0 0 0 1 0 0</td>
<td>1 0 0 1 0</td>
</tr>
<tr>
<td>0 0 0 0 0 1 1 0</td>
<td>0 1 1 1 0</td>
</tr>
<tr>
<td>0 0 0 0 0 1 0 1</td>
<td>0 1 1 1 1</td>
</tr>
<tr>
<td>0 0 0 0 0 0 1 0</td>
<td>1 1 1 0 0</td>
</tr>
<tr>
<td>0 0 0 0 0 0 1 1</td>
<td>0 0 0 0 1</td>
</tr>
<tr>
<td>0 0 0 0 0 0 0 1</td>
<td>1 1 1 0 1</td>
</tr>
</tbody>
</table>