Buletin Teknik Elektro dan Informatika
(Bulletin of Electrical Engineering and Informatics)
Vol.1, No.3, September 2012, pp. 225~232
ISSN: 2089-3191

Codes Detecting and Correcting Solid Burst Errors

Pankaj Kr. Das
Department of Mathematics, Shivaji College, University of Delhi, Raja Garden, Delhi-110027
e-mail: pankaj4thapril@yahoo.co.in

Abstract

The main tasks in coding theory are to find codes which detect and correct errors. The bounds are important in terms of error-detecting and -correcting capabilities of the codes. Solid Burst errors are the type of errors that occur in several communication channels. This paper obtains lower and upper bounds on the number of parity-check digits required for linear codes capable of detecting and correcting such errors. Illustrations of codes for detecting as well as correcting such errors are provided.

Keywords: *parity check matrix, syndrome, standard array, solid burst error*

1. Introduction

Investigations in coding theory have been made in several directions but one of the most important directions has been the detection and correction of errors. It began with Hamming codes [12] for single errors, Golay codes ([8], [9]) for double and triple random errors and thereafter BCH codes ([10], [11], [13]) were studied for multiple error correction. There is a long history towards the growth of the subject and many of the codes developed have found applications in numerous areas of practical interest. One of the areas of practical importance in which a parallel growth of the subject took place is that of burst error detecting and correcting codes. It is because of the fact that in many communication channels, burst errors occur more frequently than random errors. A burst of length b may be defined as follows:

Definition 1: A burst of length b is a vector whose only non-zero components are among some b consecutive components, the first and the last of which is non zero.

When in a burst of length b, all the b components, in which the non-zero components are confined, are non zero i.e., all the digits among the b components are in error, such type of burst is known as solid burst. Such bursts are prevalent in channels viz. semiconductor memory data[14], supercomputer storage system [2]. A solid burst may be defined as follows:

Definition 2: A solid burst of length b is a vector with non zero entries in some b consecutive positions and zero elsewhere.

Schillinger [18] developed codes that correct solid burst error. Shiva and Cheng [20] produced a paper for correcting multiple solid burst error of length b in binary code with a very simple decoding scheme. Among many, some of the good research on solid burst can be mentioned such as Bossen [3], Sharma and Dass [19], Etzion [7], Argyrides et al. [1].

It is important to know the ultimate capabilities and limitation of error correcting codes. This information, along with the knowledge of what is practically achievable, indicates which problems are virtually solved and which needs further work. This was initiated by Hamming[12] who was concerned with both code constructions and bounds. The bounds on the number of parity check symbols are important from the point of efficiency of a code. The lesser of parity check symbols in a code, the more is the rate of information of the code.

As the nature of error differs from channel to channel depending upon the behaviour of channels (or the kind of errors which occur during the process of transmission), there is a need to deal with many types of error patterns and accordingly codes are to be constructed to combat such error patterns. Though many works have been done on solid burst of length b or less, the

Received June 15, 2012; Revised August 23, 2012; Accepted September 17, 2012

bounds on the number of parity checks for linear codes over GF(q) detecting and correcting such errors are not obtained properly.

In this direction, the paper studies linear codes over GF(q) that detect and correct solid burst errors. Section 1 i.e., the Introduction gives brief view of the importance of bounds on parity check digits of a code and the requirement for consideration of solid burst errors. In section 2, we obtain the lower and upper bounds on the number of parity check digits of linear codes that detect any solid burst of length *b* or less. The section 3 presents the similar bounds for codes correcting such errors. In what follows a linear code will be considered as a subspace of the space of all *n*-tuples over GF(q). The distance between two vectors shall be considered in the Hamming sense.

2. Codes Detecting Solid Burst errors

We consider linear codes over GF(q) that are capable of detecting any solid burst error of length *b* or less. Clearly, the patterns to be detected should not be code words. In other words we consider codes that have no solid burst error of length *b* or less as a code word. Firstly, we obtain a lower bound over the number of parity-check digits required for such a code. The proof is based on the technique used in theorem 4.13, Peterson and Weldon [16].

Theorem 1. Any (n, k) linear code over GF(q) that detects any solid burst of length *b* or less must have at least $\log_q(1+b)$ parity-check digits.

Proof. The result will be proved on the basis that no detectable error vector can be a code word.

Let V be an (n, k) linear code over GF(q). Consider a set X of all those vectors such that the some fixed non-zero components are in *b* or less consecutive positions starting from the first position.

We claim that no two vectors of the set X can belong to the same coset of the standard array; else a code word shall be expressible as a sum or difference of two error vectors.
Assume on the contrary that there is a pair, say x_1, x_2 in X belonging to the same coset of the standard array. Their difference viz. $x_1 - x_2$ must be a code vector. But $x_1 - x_2$ is a vector all of whose non zero components are in *b* or less consecutive components i.e., $x_1 - x_2$ is a solid burst of length *b* or less, which is a contradiction. Thus all the vectors in X must belong to distinct cosets of the standard array. The number of such vectors over GF(q), including the vector of all zero, is clearly

$$1+b.$$

The theorem follows since there must be at least this number of cosets.

In the following theorem, an upper bound on the number of check digits required for the construction of a linear code considered in theorem 1 is provided. This bound assures the existence of a linear code that can detect all solid burst error of length *b* or less. The proof is based on the well known technique used in Varshomov-Gilbert Sacks bound by constructing a parity check matrix for such a code (refer Sacks[17], also theorem 4.7 Peterson and Weldon [16]).

Theorem 2. There exists an (n, k) linear code over GF(q) that has no solid burst of length *b* or less as a code word provided that

$$\text{n-k} > \log_q\left(\sum_{i=0}^{b-1}(q-1)^i\right)$$

Proof. The existence of such a code will be shown by constructing an appropriate $(n\text{-}k)\times n$ parity-check matrix H. The requisite parity-check matrix H shall be constructed as follows:

Select any non-zero (n-k)-tuples as the first j-1 columns h_1, h_2, ..., h_{j-1} appropriately, we lay down the condition to add j^{th} column h_j such that h_j should not be a linear sum of immediately preceding consecutive upto *b*-1 columns. In other words,

$$h_j \neq u_{j-1} h_{j-1} + u_{j-2} h_{j-2} + \ldots + u_{j-s+2} h_{j-s+2} + u_{j-s+1} h_{j-s+1} ,$$

where s ≤ b, j ≥ s, the coefficients $u_i \in GF(q)$ are non zero.

This condition ensures that no solid burst of length *b* or less will be a code word which thereby means that the code shall be able to detect solid bursts of length *b* or less. The number of ways in which the coefficients u_i can be selected, including the vector of all zeros, is

$$1+\sum_{i=1}^{b-1}(q-1)^i .$$

At worst, all these linear combinations might yield a distinct sum.
Therefore a column h_j can be added to H provided that

$$q^{n-k} > 1+\sum_{i=1}^{b-1}(q-1)^i .$$

Or,

$$n-k > \log_q\left(\sum_{i=0}^{b-1}(q-1)^i\right).$$

Example 1. Consider a (4, 2) binary code with the 2×4 matrix H which has been constructed by the synthesis procedure given in the proof of theorem 2 by taking *b* = 3, n = 4.

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

The null space of this matrix can be used to detect all solid bursts of length 3 or less. It may be verified from error pattern-syndromes table 1 that the syndromes of all solid bursts of length 3 or less are non zero.

Table 1

Error patterns	**Syndromes**
Solid bursts of length 1	
1000	10
0100	01
0010	10
0001	01
Solid bursts of length 2	
1100	11
0110	11
0011	11
Solid bursts of length 3	
1110	01
0111	10

3. Codes correcting solid burst errors

Out of the two results obtained in this section, the first result gives a lower bound on the number of check digits required for the existence of a linear code over GF(q) that corrects all solid bursts of length *b* or less. The second result gives an upper bound on the number of check digits which ensures the existence of such a code. The proof of the first result is based on the technique used in theorem 4.16, Peterson and Weldon[16]. The proof of the second result is based on the same well known technique used in Varshomov-Gilbert Sacks bound by constructing a parity check matrix for such a code (refer Sacks[17], also theorem 4.7 Peterson and Weldon [16]).

Theorem 3. An (n, k) linear code over GF(q) that corrects all solid bursts of length *b* or less must have at least:

$$\log_q\left(1+\sum_{i=1}^{b}(n-i+1)(q-1)^i\right) \text{ parity check digits.}$$

Proof. The proof is based on counting the number of correctable error vectors and comparing it with the available number of cosets.

We have,

the number of solid bursts of length 1 = $n(q-1)$.
the number of solid bursts of length 2 = $(n-1)(q-1)^2$.
the number of solid bursts of length 3 = $(n-2)(q-1)^3$.
. . . .
. . . .
the number of solid bursts of length *b* = $(n-b+1)(q-1)^b$.

So, the total number of correctable error vectors including the vector of all zero's is

$$1+\sum_{i=1}^{b}(n-i+1)(q-1)^i .$$

For correction, all these vectors must belong to different cosets. The total number of cosets available is q^{n-k} . Therefore we must have

$$q^{n-k} \geq 1+\sum_{i=1}^{b}(n-i+1)(q-1)^i .$$

Or,

$$n-k \geq \log_q\left(1+\sum_{i=1}^{b}(n-i+1)(q-1)^i\right).$$

Now what follows is an upper bound on the number of check digits required for the construction of a linear code discussed in theorem 3. This bound assures the existence of a linear code that can correct all solid burst error of length *b* or less.

Theorem 4. There shall always exist an (n, k) linear code over GF(q) that corrects all bursts of length *b* or less ($n > 2b$) provided that

$$q^{n-k} > 1+\sum_{i=1}^{b}\sum_{l=1}^{b}(n-l-i+1)(q-1)^{i+l-1}$$

Proof. The existence of such a code will be proved by constructing an $(n-k)\times n$ parity check matrix H for the desired code as follows.

Select any non zero (n-k) tuple as the first column h_1 of the matrix H. After having selected the first j-1 columns $h_1, h_2, \ldots, h_{j-1}$ appropriately, we lay down the condition to add j^{th} column h_j as follows:

h_j should not be a linear sum of immediately preceding upto *b*-1 consecutive columns $h_{j-1}, h_{j-2}, \ldots, h_{j-b+1}$, together with any *b* or fewer consecutive columns from amongst the first j-*b* columns $h_1, h_2, \ldots, h_{j-b}$ i.e.,

$$h_j \neq (u_{j-1} h_{j-1} + u_{j-2} h_{j-2} + \ldots\ldots + u_{j-s+1} h_{j-s+1} + u_{j-s} h_{j-s}) + (v_i h_i + v_{i+1} h_{i+1} + \ldots\ldots + v_{i+s'-2} h_{i+s'-2} + v_{i+s'-1} h_{i+s'-1})$$

where $u_i, v_i \in GF(q)$ are non zero coefficients, $s \le b-1$, $s' \le b$ and the columns h_i in the second bracket are any *b* or less consecutive columns among the first (j-1-s) columns.

This condition ensures that there shall not be a code vector which can be expressed as sum (difference) of two solid bursts of length *b* or less each. Thus, the coefficients u_i form a solid burst of length s and the coefficients v_i form a solid burst of length *b* or less in a (j-1-s)-tuple.

The number of choices of these coefficients can be calculated as follows:
If u_i is chosen to be a solid burst of length (*b*-1), then the number of solid bursts of length *b* or less in a (j-b)-tuple, corresponding to the coefficient v_i, is given by

$$\sum_{i=1}^{b} (j-b-i+1)(q-1)^i \qquad \text{(refer theorem 3).}$$

If u_i is chosen to be a solid burst of length (*b*-2), then the number of solid bursts of length *b* or less in a (j-b+1)-tuple, corresponding to the coefficient v_i, is given by

$$\sum_{i=1}^{b} (j-b+1-i+1)(q-1)^i .$$

Continuing the process, if u_i is chosen to be a solid burst of length 0, then the number of solid bursts of length *b* or less in a (j-1)-tuple, corresponding to the coefficient v_i, is

$$\sum_{i=1}^{b} (j-1-i+1)(q-1)^i .$$

Therefore, the total number of possible choices of the coefficients u_i and v_i is

$$(q-1)^{b-1} \sum_{i=1}^{b} (j-b-i+1)(q-1)^i + (q-1)^{b-2} \sum_{i=1}^{b} (j-b+1-i+1)(q-1)^i + \ldots\ldots + (q-1) \sum_{i=1}^{b} (j-2-i+1)(q-1)^i + \sum_{i=1}^{b} (j-1-i+1)(q-1)^i ,$$

which can be written as

$$\sum_{i=1}^{b} \sum_{l=1}^{b} (n-l-i+1)(q-1)^{i+l-1} .$$

Thus the column h_j can be added provided

$$q^{n-k} > 1 + \sum_{i=1}^{b}\sum_{l=1}^{b}(n-l-i+1)(q-1)^{i+l-1}.$$

Example 2. Consider a (9, 3) binary code with the 6×9 matrix H which has been constructed by the synthesis procedure given in the proof of theorem 4 by taking b = 3, n = 9.

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

The null space of this matrix can be used to correct all solid bursts of length 3 or less. It may be verified from error pattern-syndromes table 2 that the syndromes of all solid bursts of length 3 or less are non zero and distinct.

Table 2

Error patterns	**Syndromes**
Solid bursts of length 1	
100000000	100000
010000000	010000
001000000	001000
000100000	000100
000010000	000010
000001000	000001
000000100	100100
000000010	010010
000000001	001001
Solid bursts of length 2	
110000000	110000
011000000	011000
001100000	001100
000110000	000110
000011000	000011
000001100	100101
000000110	110110
000000011	011011
Solid bursts of length 3	
111000000	111000
011100000	011100
001110000	001110
000111000	000111
000011100	100111
000001110	110111
000000111	111111

4. Discussion and Conclusion

This paper presents the bounds on parity checks for codes capable of detecting and correcting solid burst errors, also deals with the construction of such codes. The bounds will determine the capability of error-detecting and -correcting of the codes. The papers ([4], [5], [6]) obtain bounds for codes dealing with **burst error**. The bounds derived in this paper are exclusively for **solid burst error**. If the types of errors occurred are known to be solid burst, these bounds will be more useful.

The optimal codes are useful from application point of view in communication as having minimum redundancy and improving the rate of transmission. Therefore optimal codes that correct all solid burst errors of length *b* or less and no other errors can be good work. Bounds similar to the ones obtained in this paper w.r.t. the metric studied by Kitakami et al. [15] may also be derived.

Acknowledgement

The author is thankful to Dr. Vinod Tyagi and Prof. B. K. Dass, University of Delhi for their valuable suggestions.

References

[1] Argyrides, C.A., Reviriego, P., Pradhan, D.K., Maestro, J.A., Matrix-Based Codes for Adjacent Error Correction, *IEEE Transactions on Nuclear Science*, Vol. 57, Iss. 4, pp. 2106-2111, 2010.
[2] Arlat, J. and Carter, W.C., Implementation and Evaluation of a (b,k)-Adjacent Error-Correcting/Detecting Scheme for Supercomputer Systems, *IBM J. RES. DEVELOP.*, Vol. 28, No. 2, pp. 159-169, 1984.
[3] Bossen, D.C., *b*-Adjacent Error Correction, *IBM Journal of Research and Development*, Vol. 14, Issue 4, pp. 402-408, 1970.
[4] Dass, B.K., A sufficient bound for codes correcting bursts with weight constraints, *Journal of the Association for Computing Machinery (J. ACM),* Vol. 22, No. 4, pp. 501-503, 1975.
[5] Dass, B.K., On a burst-error correcting code, Journal of Information and Optimization Sciences, Vol. 1, No. 3, pp. 291-295, 1980.
[6] Dass, B.K., Low-density burst error correcting linear codes, *Adv. in Mngt. Studies,* Vol. 2, No. 4, pp. 375-385, 1983.
[7] Etzion, T., Optimal codes for correcting single errors and detecting adjacent errors, *IEEE Trans. Inform. Theory*, vol. 38, pp. 1357–1360, July, 1992.
[8] Golay, M.J.E., Notes on digital coding, *Proc. IRE*, 37, 657, 1949.
[9] Golay, M.J.E., Binary Coding, *IRE Trans., PGIT-4*, pp. 23-28, 1954.
[10] Golay, R.C. and D.K.Ray-Chaudhuri, On a class of Error Correcting Group Codes, *Inf. and Control*, 3, pp. 68-79, 1960.
[11] Golay, R.C. and D.K.Ray-Chaudhuri, Further Results on Error Correcting Binary Group Codes, *Inf. and Control*, 3, pp. 279-290, 1960.
[12] Hamming, R.W., Error-detecting and error-correcting codes, *Bell System Technical Journal*, Vol. 29, pp. 147-160, 1950.
[13] Hocquenghem, A., Codes Corecteurs d'Erreurs, *Chiffres*, Vol. 2, pp. 147-156, 1959.
[14] Jensen, D.W., Block code to efficiently correct adjacent data and/or check bit errors, Patent no: US 6604222 B1, Date of Patent Aug 5,*(www.google.com/patents/US6604222)*, 2003.
[15] Kitakami, M., Jiang, S. and Fujiwara, E., Metrics of error locating codes, *Proc. of 1997 Int. Symp. Infom. Theory*, pp. 264, 1997.
[16] Peterson, W.W. and Weldon, E.J., Jr., *Error-Correcting Codes*, 2nd edition, The MIT Press, Mass, 1972.
[17] Sacks, G.E., Multiple error correction by means of parity-checks, *IRE Trans. Inform. Theory IT*, Vol. 4, pp. 145--147, 1958.
[18] Schillinger, A.G., A class of solid burst error correcting codes, *Polytechnic Institute of Brooklyn* , N.Y., Research Rept. PIBMRI, April, pp. 1223-64, 1964.
[19] Sharma, B.D. and Dass, B.K., Adjacent error correcting binary perfect codes, *J. Cybernetics*, No. 7, pp. 9-13, 1977.
[20] Shiva, S.G.S. and Sheng, C.L., Multiple solid burst-error-correcting binary codes, *IEEE Trans. Inform. Theory*, IT-15, pp. 188-189, 1969.