www.combinatorics.ir

Transactions on Combinatorics
ISSN (print): 2251-8657, ISSN (on-line): 2251-8665
Vol. 4 No. 2 (2015), pp. 45-55.

www.ui.ac.ir

LINEAR CODES ON SOLID BURSTS AND RANDOM ERRORS

PANKAJ KUMAR DAS

Communicated by Dianhua Wu

Abstract. The paper presents lower and upper bounds on the number of parity check digits required for a linear code that detects solid bursts of length b or less and simultaneously any e or less random errors. An example of such a code is also provided. Further, codes capable of detecting and simultaneously correcting such errors have also been dealt with.

1. **Introduction**

The very purpose of coding theory is to detect and correct errors that occur during transmission of data from one place to another over noisy channel. The nature of errors depends on the behavior of the communication channel. Solid burst type of errors are often found in storage channels (viz. semiconductor memory data [12], supercomputer storage system [2]). Thus, codes are required to construct to deal with such type of errors. A solid burst may be defined as follows:

Definition 1.1. *A solid burst of length b is a vector with non zero entries in some b consecutive positions and zero elsewhere.*

Some of the worth mentioning works on solid burst are due to Schillinger [16], Shiva and Cheng [18], Bossen [3], Sharma and Dass [17], Etzion [10], Argyrides et al.[1]. For some more study on bounds of codes dealing with solid burst errors, one may refer to [4, 5, 6, 7, 8].

The bounds on the number of parity check symbols of a linear code 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

MSC(2010): Primary: 94B65, Secondary: 94B05, 94B20.

Keywords: Parity check matrix, syndrome, solid burst error.

Received: 26 December 2013, Accepted: 2 September 2014.

of the code. In fact, the bounds on parity check digits will not only let us know the ultimate capabilities and limitation of error detecting/correcting codes and but also let us know which problems are virtually solved and which needs further work. Hamming [11] was the first who was concerned in this regard and he studied bounds on codes and their constructions.

Das [4] obtains bounds for codes detecting solid burst of length b or less. It is quite possible that the storage channel which is interfered by solid burst error may be encountered with some small number of random errors. Therefore, while it is all important to consider the detection/correction of solid burst, but care should also be taken to handle the detection/correction of some random errors. The present paper gives an attempt in this direction and obtain bounds for codes detecting /and simultaneously correcting solid burst of length b or less and any e or less random errors.

It should be noted that similar study was also presented in the paper [9]. In the paper [9], Dass and Muttoo obtained upper bound on codes that detect random errors or closed-loop low-density bursts. They had not studied the case when closed-loop low-density bursts as well as random errors occur at the same time. The present paper takes the case of solid burst and random error, and gives a study when *solid bursts as well as random errors occur simultaneously.* If the error occurred during communication is in the form of solid burst which is prevalent in many storage channels, then we need to consider only this pattern rather than considering other patterns of errors (say bursts or closed-loop bursts) and wasting the capacity of the system by detecting/correcting non-errors by default.

The present paper is organized as follows. 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 error and random error simultaneously. In Section 2, we obtain lower and upper bounds on the number of parity check digits of linear codes that detect solid burst of length b or less and simultaneously any random e or less error ($e < b$). An example is also provided in this section. Section 3 deals with codes that are capable of detecting and simultaneously correcting such errors.

2. Codes Detecting Solid Burst and Random Errors

We consider linear codes over GF(q) that detect any solid bursts of length b or less and simultaneously any random e or less error ($e < b$). Clearly, the patterns to be detected should not be code words. In other words, we consider codes that have no solid burst of length b or less and simultaneously any random e or less error ($e < b$) as a code word. In the following, 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 [13].

Theorem 2.1. *The number of parity check digits for an (n,k) linear code over GF(q) that detects any solid burst of length b or less and simultaneously any random e or less error $(e<b)$ is at least*

$$\log\left[1+b\sum_{i=0}^{\lfloor\frac{e}{2}\rfloor}\binom{n-b-1}{i}(q-1)^i\right].$$

Proof. The result will be proved on the fact that no detectable error vector can be a code word. Let V be an (n,k) linear code over GF(q) and $\mathcal{X}$ be a set of all those vectors such that the some fixed non-zero component are confined consecutively from the first position and may go up to first b positions, and the last $n-b-1$ positions may have at most $\lfloor\frac{e}{2}\rfloor$ any non zero component.

We claim that any two vectors of the set $\mathcal{X}$ can not 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. If possible, we assume the contrary that there is a pair; say x_1, x_2 in $\mathcal{X}$ belonging to the same coset of the standard array. Then their difference 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 in the first b positions and at most $2\lfloor\frac{e}{2}\rfloor(\leq e)$ positions in the remaining $n-b-1$ positions. This means x_1-x_2 is an error vector and is a code vector. This is not possible. Thus, all the vectors in $\mathcal{X}$ 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

$$1+b\sum_{i=0}^{\lfloor\frac{e}{2}\rfloor}\binom{n-b-1}{i}(q-1)^i.$$

Since the maximum available number of cosets is q^{n-k}, we have

$$q^{n-k}\geq 1+b\sum_{i=0}^{\lfloor\frac{e}{2}\rfloor}\binom{n-b-1}{i}(q-1)^i$$

$$\Rightarrow n-k\geq\log\left[1+b\sum_{i=0}^{\lfloor\frac{e}{2}\rfloor}\binom{n-b-1}{i}(q-1)^i\right].$$

□

Now the following theorem gives an *upper* bound on the number of check digits required for the construction of a linear code considered in Theorem 2.1. This bound assures the existence of a linear code that can detect all solid bursts of length b or less and simultaneously any random e or less error $(e<b)$. 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 [14], also Theorem 4.7 Peterson and Weldon [13]).

Theorem 2.2. *There exists an (n,k) linear code over GF(q) that detects any solid burst of length b or less and simultaneously any random e or less error $(e<b)$ provided that*

$$q^{n-k} > \sum_{i=0}^{e-1}\binom{n-1}{i}(q-1)^i + \sum_{s=e+1}^{b}\left\{\sum_{i=0}^{e}\binom{n-s-1}{i}(q-1)^{i+s-1} + (n-s-1)\sum_{i=0}^{e-1}\binom{n-s-3}{i}(q-1)^{i+s}\right\}. \tag{2.1}$$

Proof. The theorem is proved by constructing an appropriate $(n-k)\times n$ parity-check matrix H for the existence of the code. The requisite parity-check matrix H shall be constructed as follows:

Select any non-zero $(n-k)$-tuples as the first $j-1$ columns $h_1, h_2, \ldots, h_{j-1}$ appropriately, we lay down the conditions to add j^{th} column h_j as follows:

Case (I) Since the code detects any e or less random errors, h_j should not be a linear combination of previous any $e-1$ or less columns, i.e.,

$$h_j \neq (u_{j-1}h_{j-1} + u_{j-2}h_{j-2} + \cdots + u_2h_2 + u_1h_1), \tag{2.2}$$

where $u_i \in GF(q)$ are any $e-1$ non zero coefficients.

This condition ensures that the code detects any e or less random errors. The number of ways in which such coefficients u_i out of $j-1$ coefficients on R.H.S. of (2.2) may be chosen is given by

$$\sum_{i=0}^{e-1}\binom{j-1}{i}(q-1)^i.$$

Case (II) Since the code detects any e or less random errors and simultaneously any solid burst of length greater than e, but less than or equal to $b(b>e)$, so

Subcase (i). h_j should not be a linear sum of immediately preceding consecutive $s-1$ $(e+1\le s\le b)$ columns, together with any linear combination of e or less columns from the last $j-s-1$ columns. In other words,

$$\begin{aligned} h_j \neq\ & (u_{j-1}h_{j-1} + u_{j-2}h_{j-2} + \cdots + u_{j-s+2}h_{j-s+2} + u_{j-s+1}h_{j-s+1}) \\ +\ & (v_{j-s-1}h_{j-s-1} + v_{j-s-2}h_{j-s-2} + \cdots + v_2h_2 + v_1h_1), \end{aligned} \tag{2.3}$$

where $e+1\le s\le b$, the coefficients $u_i \in GF(q)$ are non zero and the coefficients $v_i \in GF(q)$ are such that any e or less coefficients are non zero.

This condition ensures that the code detects all solid bursts of length $s(e+1\le s\le b)$ (the last component being in solid burst error) and any e or less random errors $(e<b)$.

The number of choices of these coefficients u_i and v_i can be calculated as follows:

If $s=e+1$, then there are e terms having u_i as coefficients in the first bracket of R.H.S. of (2.3). The number of coefficients v_i out of $j-s-1$ coefficients may be chosen by $\sum_{i=0}^{e}\binom{j-e-2}{i}(q-1)^i$ ways.

So total number of combinations for $s = e + 1$ is given by

$$(q-1)^e \sum_{i=0}^{e} \binom{j-e-2}{i} (q-1)^i.$$

If $s = e + 2$, then there are $e + 1$ terms having u_i as coefficients in the first bracket of R.H.S. of (2.3). The number of coefficients v_i out of $j - s - 1$ coefficients is by $\sum_{i=0}^{e} \binom{j-e-3}{i} (q-1)^i$. So, total number of combinations for $s = e + 2$ is given by

$$(q-1)^{e+1} \sum_{i=0}^{e} \binom{j-e-3}{i} (q-1)^i.$$

For $s = b$, then there will be $b-1$ terms in the first bracket on R.H.S. of (2.3) which can be chosen by $(q-1)^{b-1}$ ways . And the coefficients v_i out of $j-s-1$ coefficients can be selected by $\sum_{i=0}^{e} \binom{j-b-1}{i} (q-1)^i$ ways. So, the total number of combinations on R.H.S. of (2.3) for $s = b$ is

$$(q-1)^{b-1} \sum_{i=0}^{e} \binom{j-b-1}{i} (q-1)^i.$$

Thus, the total number of choices of these coefficients u_i and v_i for the *subcase (i)* is

$$\sum_{s=e+1}^{b} \left\{ (q-1)^{s-1} \sum_{i=0}^{e} \binom{j-s-1}{i} (q-1)^i \right\}$$
$$= \sum_{s=e+1}^{b} \sum_{i=0}^{e} \binom{j-s-1}{i} (q-1)^{i+s-1}.$$

Subcase (ii). h_j should not be a linear sum of previous any linear combination of e or less columns, along with any linear sum of s consecutive columns ($e + 1 \leq s \leq b$), not including h_j (with the condition that not more than s consecutive columns occur), i.e.,

$$\begin{aligned} h_j &\neq (u_{j-1}h_{j-1} + u_{j-2}h_{j-2} + \cdots + u_{j-l}h_{j-l} + u_{j-l}h_{j-l}) \\ &+ (v_{j-l-1}h_{j-l-1} + v_{j-l-2}h_{j-l-2} + \cdots + v_2h_2 + v_1h_1), \end{aligned} \tag{2.4}$$

where $1 \leq l \leq j - e - 2$, u_{j-l} is zero coefficient, v_i's are such that the number of consecutive non zero v_i's is s and if v_t and v_m is the first and last of such consecutive nonzeros, then the v_i's (before v_{t-1} and after v_{m+1}) together with u_i's can be at most $e - 1$ non zero.

This condition ensures that the code detects all solid bursts of length $s(e + 1 \leq s \leq b)$ (the last component not being in solid burst error) and any e or less random errors ($e < b$).

The coefficients u_i and v_i of the above expression (2.4) can be chosen in the following ways (as calculated in *subcase (i)*):
For $s = e + 1$, the number of linear combinations of coefficients on the R.H.S. of (2.4) is

$$(j-e-2)(q-1)^{e+1} \sum_{i=0}^{e-1} \binom{j-e-4}{i} (q-1)^i.$$

For $s = e+2$, the number of linear combinations of coefficients on the R.H.S. of (2.4) is

$$(j-e-3)(q-1)^{e+2}\sum_{i=0}^{e-1}\binom{j-e-5}{i}(q-1)^i.$$

For $s = b$, the number of linear combinations of coefficients on the R.H.S. of (2.4) is

$$(j-b-1)(q-1)^{b}\sum_{i=0}^{e-1}\binom{j-b-3}{i}(q-1)^i.$$

So, the total number of linear combinations of coefficients u_i and v_i for the *subcase (ii)* is

$$\begin{aligned}&\sum_{s=e+1}^{b}(j-s-1)(q-1)^s\sum_{i=0}^{e-1}\binom{j-s-3}{i}(q-1)^i\\ &= \sum_{s=e+1}^{b}(j-s-1)\sum_{i=0}^{e-1}\binom{j-s-3}{i}(q-1)^{i+s}.\end{aligned}$$

Therefore, the number of possible linear combinations of coefficients on the R.H.S. of (2.2), (2.3) and (2.4), including the vector of all zeros, is

$$\sum_{i=0}^{e-1}\binom{j-1}{i}(q-1)^i+\sum_{s=e+1}^{b}\sum_{i=0}^{e}\binom{j-s-1}{i}(q-1)^{i+s-1}+\sum_{s=e+1}^{b}(j-s-1)\sum_{i=0}^{e-1}\binom{j-s-3}{i}(q-1)^{i+s}$$

$$=\sum_{i=0}^{e-1}\binom{j-1}{i}(q-1)^i+\sum_{s=e+1}^{b}\left\{\sum_{i=0}^{e}\binom{j-s-1}{i}(q-1)^{i+s-1}+(j-s-1)\sum_{i=0}^{e-1}\binom{j-s-3}{i}(q-1)^{i+s}\right\}.$$

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

$$\begin{aligned}q^{n-k} \;>\; &\sum_{i=0}^{e-1}\binom{j-1}{i}(q-1)^i\\ +\; &\sum_{s=e+1}^{b}\left\{\sum_{i=0}^{e}\binom{j-s-1}{i}(q-1)^{i+s-1}+(j-s-1)\sum_{i=0}^{e-1}\binom{j-s-3}{i}(q-1)^{i+s}\right\}.\end{aligned}\tag{2.5}$$

Replacing j by n gives the theorem. □

Alternative Form

If n is the largest positive integer for which the inequality (2.5) is true, then replacing j by $n+1$, the inequality (2.5) gets reversed and the inequality becomes

$$\begin{aligned}q^{n-k} \;\le\; &\sum_{i=0}^{e-1}\binom{n}{i}(q-1)^i\\ &+\sum_{s=e+1}^{b}\left\{\sum_{i=0}^{e}\binom{n-s}{i}(q-1)^{i+s-1}+(n-s)\sum_{i=0}^{e-1}\binom{n-s-2}{i}(q-1)^{i+s}\right\}.\end{aligned}\tag{2.6}$$

Asymptotic Form

We deduce the asymptotic form of the above inequality (2.6) over GF(2). By taking $q = 2$, the inequality (2.6) becomes

$$2^{n-k} \leq \sum_{i=0}^{e-1}\binom{n}{i} + \sum_{s=e+1}^{b}\left\{\sum_{i=0}^{e}\binom{n-s}{i} + (n-s)\sum_{i=0}^{e-1}\binom{n-s-2}{i}\right\}.$$

or,

$$\begin{aligned}(2.7)\qquad 2^{n-k} \;\leq\; & \sum_{i=(\frac{n-s+1}{n})n}^{n}\binom{n}{i} \\ & + \sum_{s=e+1}^{b}\left\{\sum_{i=(\frac{n-s-e}{n-s})(n-s)}^{n-s}\binom{n-s}{i} + (n-s)\sum_{i=(\frac{n-s-e-1}{n-s-2})(n-s-2)}^{n-s-2}\binom{n-s-2}{i}\right\}.\end{aligned}$$

From the Chernov Bound,

$$\sum_{i=\alpha n}^{n}\binom{n}{i} \leq \alpha^{-\alpha n}\beta^{-\beta n}, \quad \beta = 1-\alpha, \quad \alpha > \frac{1}{2},$$

we can deduce the inequality (2.7) as follows:

$$\begin{aligned}(2.8)\ 2^{n-k} \;\leq\; & \left(\frac{n-s+1}{n}\right)^{-(n-s+1)}\left(\frac{s-1}{n}\right)^{-(s-1)} + \sum_{s=e+1}^{b}\left\{\left(\frac{n-s-e}{n-s}\right)^{-(n-s-e)}\left(\frac{e}{n-s}\right)^{-e}\right. \\ & \left. +(n-s)\left(\frac{n-s-e-1}{n-s-2}\right)^{-(n-s-e-1)}\left(\frac{e-1}{n-s-2}\right)^{-(e-1)}\right\},\end{aligned}$$

where $n > max\{2(s-1), s+2e\}$. We know the binary entropy function is given by

$$H(p) = -p\log_2 p - (1-p)\log_2(1-p).$$

Therefore, the inequality (2.8) reduces to

$$2^{n-k} \leq 2^{nH(\frac{n-s+1}{n})} + \sum_{s=e+1}^{b}\left\{2^{(n-s)H(\frac{n-s-e}{n-s})} + (n-s)2^{(n-s-2)H(\frac{n-s-e-1}{n-s-2})}\right\}.$$

Example 2.1. For a $(9, 2)$ linear code over $GF(2)$, we construct the following 7×9 parity check matrix H, according to the synthesis procedure given in the proof of Theorem 2.2 by taking $e = 2$, $b = 4$.

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

The null space of this matrix can be used to correct all solid bursts of length 4 or less and simultaneously any random 2 or less errors. It may be verified from error pattern-syndromes Table 2.1 that the syndromes of all solid bursts of length 4 or less and simultaneously any random 2 or less errors are non zero, showing thereby that the code that is the null space of this matrix can detect all solid bursts of length 4 or less and simultaneously any random 2 or less errors.

Table 2.1

Error pattern - syndromes Table

Error-patterns	Syndromes	Error-patterns	Syndromes	Error-patterns	Syndromes
100000000	1000000	000001100	0000011	001110011	1100011
010000000	0100000	000001010	1010111	000111000	0001110
001000000	0010000	000001001	0101000	000111010	1011011
000100000	0001000	000000110	1010100	000111001	0100100
000010000	0000100	000000101	0101011	000111011	1110001
000001000	0000010	000000011	1111111	000011100	0000111
000000100	0000001	111000000	1110000	000011101	0101101
000000010	1010101	111010000	1110100	000001110	1010110
000000001	0101010	111001000	1110010	000000111	1111110
110000000	1100000	111000100	1110001	111100000	1111000
101000000	1010000	111000010	0100101	111101000	1111010
100100000	1001000	111000001	1011010	111100100	1111001
100010000	1000100	111011000	1110110	111100010	0101101
100001000	1000010	111010100	1110101	111100001	1010010
100000100	1000001	111010010	0100001	111101100	1111011
100000010	0010101	111010001	1011110	111101010	0101111
100000001	1101010	111001100	1110011	111101001	1010000
011000000	0110000	111001010	0100111	111100110	0101100
010100000	0101000	111001001	1011000	111100101	1010011
010010000	0100100	111000110	0100100	111100011	0000111
010001000	0100010	111000101	1011011	011110000	0111100
010000100	0100001	111000011	0001111	011110100	0111101
010000010	1110101	011100000	0111000	011110010	1101001
010000001	0001010	011101000	0111010	011110001	0010110

cont...

Error-patterns	Syndromes	Error-patterns	Syndromes	Error-patterns	Syndromes
001100000	0011000	011100100	0111001	011110110	1101000
001010000	0010100	011100010	1101101	011110101	0010111
001001000	0010010	011100001	0010010	011110011	1000011
001000100	0010001	011101100	0111011	001111000	0011110
001000010	1000101	011101010	1101111	001111010	1001011
001000001	0111010	011101001	0010000	001111001	0110100
000110000	0001100	011100110	1101100	001111011	1100001
000101000	0001010	011100101	0010011	000111100	0001111
000100100	0001001	011100011	1000111	000111101	0100101
000100010	1011101	001110000	0011100	000011110	1010010
000100001	0100010	001110100	0011101	000001111	1111100
000011000	0000110	001110010	1001001		
000010100	0000101	001110001	0110110		
000010010	1010001	001110110	1001000		
000010001	0101110	001110101	0110111		

3. Simultaneous Detection and Correction of Solid Burst and Random Errors

This section determines extended Reiger's bound (refer [15]; also Theorem 4.15, Peterson and Weldon [13]) for simultaneous detection and correction of solid burst of length b or less and simultaneously any random e or less error ($e < b$). The following theorem gives a bound on the number of parity-check digits for a linear code that simultaneously detects and corrects such errors.

Theorem 3.1. *The number of parity check symbols in an (n, k) linear code over GF(q) that corrects any solid burst of length b or less and simultaneously any random e or less error $(e < b)$ must have at least*

$$\log\left[1 + 2b\sum_{i=0}^{e}\binom{n-2b-1}{i}(q-1)^i\right].$$

Further, if the code corrects all solid bursts of length b or less and simultaneously any random e or less error $(e < b)$, and simultaneously detects any solid burst of length d or less and simultaneously any random f or less error $(f < d, b < d, e < f)$, then the number of parity-check digits of the code is *at least*

$$\log\left[1+(b+d)\sum_{i=0}^{\lfloor\frac{e+f}{2}\rfloor}\binom{n-(b+d)-1}{i}(q-1)^i\right].$$

Proof. For the first part, consider a vector that has the form of a solid burst of length $2b$ or less together with any $2e$ or less random error. The vector can be expressible as a sum or difference of two vectors, each of which is a solid burst of length b or less together with any random e or less error ($e < b$). These component vectors must belong to different cosets of the standard array, because both such errors are correctable errors. Accordingly, such a vector viz. a solid burst of length $2b$ or less together with any random $2e$ or less error can not be a code vector. In view of Theorem 2.1, the number of parity check digits, such a code must have, is at least

$$\log\left[1+2b\sum_{i=0}^{e}\binom{n-2b-1}{i}(q-1)^i\right].$$

Further, for the second part, consider a vector which has the form of a solid burst of length $(b+d)$ or less together with any $(e+f)$ or less random error. Such a vector is expressible as a sum or difference of two vectors, one of which has the form of a solid burst of length b or less together with any random e or less error ($e < b$) and the other is a solid burst of length d or less together with any random f or less error ($f < d$). Both such component vectors, one being a detectable error and the other being a correctable error, can not belong to the same coset of the standard array. Therefore, such a vector can not be a code vector, i.e., a vector which is a solid burst of length $(b+d)$ or less together with any $(e+f)$ or less random error can not be a code vector. Hence, by Theorem 2.1, the number of parity check digits that code must have is at least

$$\log\left[1+(b+d)\sum_{i=0}^{\lfloor\frac{e+f}{2}\rfloor}\binom{n-(b+d)-1}{i}(q-1)^i\right].$$

□

Acknowledgments

The author would like to thank the referee(s) for careful review of the manuscript and helpful comments.

References

[1] C. A. Argyrides, P. Reviriego, D. K. Pradhan and J. A. Maestro, Matrix-Based Codes for Adjacent Error Correction, *IEEE T. NUCL. SCI.*, **57** no. 4 (2010) 2106-2111.

[2] J. Arlat and W. C. Carter, Implementation and Evaluation of a (b,k)-Adjacent Error-Correcting/Detecting Scheme for Supercomputer Systems, *IBM J. Res. Develop.*, **28** no. 2 (1984) 159-169.

[3] D. C. Bossen, *b*-Adjacent Error Correction, *IBM J. Res. Develop.*, **14** no. 4 (1970), 402-408.

[4] P. K. Das, Codes Detecting and Correcting Solid Burst Errors, *Bulletin of Electrical Engineering and Informatics*, **1** no. 3 (2012) 225-232.

[5] P. K. Das, Codes Detecting and Locating Solid Burst Errors, *Rom. J. Math. Comput. Sci.*, **2** no. 2 (2012) 1-10.

[6] P. K. Das, Blockwise Solid Burst Error Correcting Codes, *International Journal on Information Theory*, **1** no. 1 (2012) 11-17.

[7] P. K. Das, Codes correcting and simultaneously detecting solid burst errors, *Com. Eng. App-Journal*, **2** no. 1 (2013) 143-150.

[8] P. K. Das, (1,3) Optimal Linear Codes on Solid Bursts, *International Journal of Information Security Science*, **1** no. 4 (2012) 106 -111.

[9] B. K. Dass and S. K. Muttoo, A bound for random and closed loop burst errors detecting codes, *Inform. Sci.*, **20** (1980) 151-156.

[10] T. Etzion, Optimal codes for correcting single errors and detecting adjacent errors, *IEEE Trans. Inform. Theory*, **38** (1992) 1357-1360.

[11] R. W. Hamming, Error-detecting and error-correcting codes, *Bell System Tech. J.*, **29**(1950) 147-160.

[12] D. W. Jensen, *Block code to efficiently correct adjacent data and/or check bit errors*, Patent no: US 6604222 B1, Date of Patent Aug 5, http://www.google.com/patents/US6604222, 2003.

[13] W .W. Peterson and E. J. Weldon(Jr.), *Error-Correcting Codes*, Second edition, The M. I. T. Press, Cambridge, Mass.-London 1972.

[14] G. E. Sacks, Multiple error correction by means of parity-checks, *IRE Trans.*, **IT-4** (1958) 145-147.

[15] S. H. Reiger, Codes for the correction of clustered errors, *Trans. IRE*, **IT-6** (1960) 16-21.

[16] A. G. Schillinger, *A class of solid burst error correcting codes*, Polytechnic Institute of Brooklyn, N. Y., Research Rept. PIBMRI, April 1964.

[17] B. D. Sharma and B. K. Dass, Adjacent error correcting binary perfect codes, *J. Cybernetics*, **7** (1977) 9-13.

[18] S. G. S. Shiva and C. L. Sheng, Multiple solid burst-error-correcting binary codes, *IEEE Trans. Information Theory*, **IT-15** (1969) 188-189.

Pankaj Kumar Das

Department of Mathematics, Shivaji College (University of Delhi), Raja Garden, Ring Road, New Delhi-110027, India

`Email: pankaj4thapril@yahoo.co.in`