A Novel Approach for Evaluating the Performance of SPC Product Codes Under Erasure Decoding

Maan A. Kousa, *Member, IEEE*

***Abstract*—Product codes are powerful codes that can be used to correct errors or recover erasures. The simplest form of a product code is that where every row and every column is terminated by a single parity bit, referred to as single parity check (SPC) product code. This code has a minimum distance of four and is thus guaranteed to recover all single, double, and triple erasure patterns. Judging the code performance based on its minimum distance is very pessimistic because the code is actually capable of recovering many higher erasure patterns. This paper develops a novel approach for deriving an upper bound on the post-decoding erasure rate for the SPC product code with iterative decoding. Simulation shows that the derived bound is very tight.**

***Index Terms*—Erasure decoding, product codes, single parity check (SPC) codes.**

I. INTRODUCTION

THE single parity check (SPC) code is one of the most popular error detection codes because it is easy to implement. In these codes, the encoder appends one bit to a sequence of $n-1$ information bits such that the resultant n-bit codeword has an even number of ones (i.e., mod-2 addition is zero).

Two SPC codes can be used jointly to construct a product code. An SPC product code has a minimum distance of 4 and can be used for error correction and/or error detection. In spite of their relatively poor minimum distance, SPC product codes have been proven to exhibit an exceedingly good performance in terms of bit error probability. The work in [1] and [2] examined the weight distribution of the codes, while [3]–[5] studied their bit error rate (BER) performance under various decoding schemes.

In some communication systems, the receiver is faced with erasures rather than, or in addition to, errors. A typical example is the ATM network where an error in the header of the cell causes the cell to be discarded. A cell may also be discarded because of buffer overflow.

SPC codes and SPC product codes have been proposed for cell loss recovery in ATM networks. Simple cell loss recovery schemes based on SPC codes are found in [6]–[9]. The authors in [10] extend the encoding to two dimensions, thus forming a product code. The analysis of the cell loss rate for the two-dimensional (2-D) case is tedious and, as a result, the authors in [10] resort to deriving an upper bound on the code performance.

Paper approved by R. D. Wesel, the Editor for Coding and Communication Theory of the IEEE Communications Society. Manuscript received August 17, 2000; revised February 18, 2001. This work was supported by King Fahd University of Petroleum and Minerals.

The author is with the Electrical Engineering Department, King Fahd University of Petroleum and Minerals, Dhahran 31261, Saudi Arabia (e-mail: makousa@kfupm.edu.sa).

Publisher Item Identifier S 0090-6778(02)00518-4.

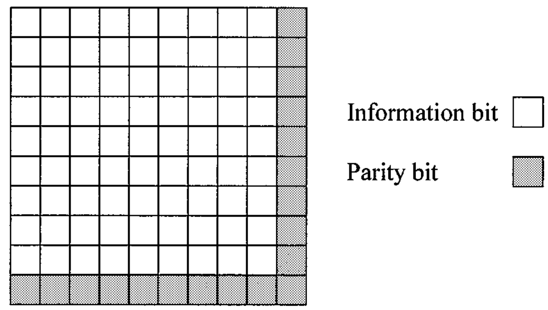

Fig. 1. The structure of SPC product code.

In this paper we derive a tight upper bound of the post-decoding erasure rate of the SPC product code. The derived bound is based on a novel approach of analyzing the structure of the erasure patterns, which leads to identifying the recoverable and the unrecoverable patterns.

The paper is organized as follows. In Section II, the encoder, decoder, and performance measure of the SPC product are described. The novel approach for analyzing the structure of erasure patterns is introduced in Section III. Recoverability study of the various erasure patterns is provided in Section IV. Results and discussions are presented in Section V, and the main conclusions are summarized in Section VI.

II. SPC PRODUCT CODE

The general structure of the SPC product code is shown in Fig. 1. It is assumed that the codeword (an $M \times N$ matrix) is received with some of the bits erased.

The decoder will perform iterative row-wise and column-wise decoding to recover the erased bits. When a single bit is erased in a row or column, it can be recovered by simple parity checking (mod-2 addition). If more than one bit is erased in a row/column, that row/column is skipped. Decoding will be performed in many rounds until all erasures are recovered or no further recovery is possible.

Clearly, the SPC product code can recover all single-, double-, and triple- erasure patterns. Patterns consisting of more than three erasures may or may not be recoverable. A pattern is unrecoverable if and only if it contains a subpattern such that every occupied row (row with erasures) and every occupied column in that subpattern has at least two erasures.

The code is evaluated in terms of the post-decoding erasure rate, P. For i erasures in a matrix of $M \times N$, let U_i denote the number of i-erasure unrecoverable patterns and e_i denote the average number of remaining erasures after decoding an unrecoverable pattern. Obviously, $U_1 = U_2 = U_3 = 0$. Then

$$P = \frac{1}{MN} \sum_{i=4}^{MN} e_i U_i p^i (1-p)^{MN-i} \tag{1}$$

where erasures are assumed to occur randomly at a rate of p.

The simplest and most straightforward approach to evaluate (1) is to base the code capability of erasure decoding on its minimum distance, that is, to assume that any pattern of four or more erasures is unrecoverable. Also, it is usually assumed that when a pattern is unrecoverable it is left intact by the decoder. Under these assumptions, (1) reduces to the following bound:

$$P < \frac{1}{MN} \sum_{i=4}^{MN} i \binom{MN}{i} p^i (1-p)^{MN-i}. \tag{2}$$

The bound in (2) is very loose, firstly, because many of the i-erasure patterns $(i > 3)$ are recoverable, and therefore $U_i \ll \binom{MN}{i}$. Second, when an i-erasure pattern is truly unrecoverable, some of its erasures may be recovered, that is, $e_i \leq i$.

The authors in [10] evaluated U_i for $i = 4$ and 5, and assumed that all patterns of six or more patterns are unrecoverable. Their bound, although significantly tighter than (2), is yet loose.

The objective of this paper is to find a tight upper bound for the expression in (1). The derivation is carried out on two lines:

1) To identify and enumerate a subset R_i of the recoverable erasure patterns for $i > 3$, leading to an upper limit on U_i.
2) For those patterns of i erasures not in R_i to calculate the average number of unrecoverable erasures e_i.

To achieve this goal, we are proposing a novel approach for analyzing the structure of erasures in the matrix. The basic idea is to define some special patterns of erasures (called *basic* patterns) from which all patterns can be generated. We will then show that this classification leads to a systematic way of enumerating the recoverable patterns for a wide range of i.

III. Structural Analysis of Erasure Patterns

The key element of our analysis is the *basic pattern*. The basic patterns are those patterns that satisfy the following two criteria:

1) The erasures in each column are consecutive (no gaps), starting from the first row.
2) The erasures in each row are consecutive, starting from the first column.

Following the above definition, a basic pattern is uniquely characterized by the number of occupied columns and the number of erasures in each of these columns. Fig. 2(a) shows some examples of basic patterns of size six.

Every basic pattern generates a set of patterns called *generated patterns*. The generated patterns are generated from a basic pattern by allowing the occupied columns and their erasures to move everywhere possible in the matrix *provided* that the number of the occupied columns and the number of erasures in each column are preserved. A demonstration is shown in Fig. 2(b).

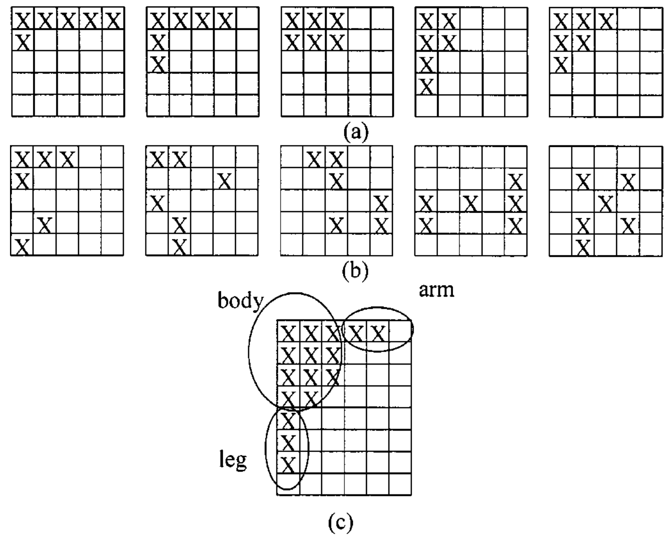

Fig. 2. Illustrative schematics of the proposed approach. (a) Examples of basic patterns of size 6. (b) Examples of patterns generated from the fifth basic pattern in (a). (c) The general structure of an unrecoverable basic pattern (UBP).

Let the number of i-erasure basic patterns be f_i. The number of generated patterns, G_m, obtained from a basic pattern m, $m = 1, 2, \ldots, f_i$, depends on the number of erasures, i, in the basic pattern, the matrix size, and the shape of the basic pattern. For a basic pattern m, the number of generated patterns G_m can be written as

$$\boldsymbol{G}_m = \boldsymbol{T}_{m,1} \times \boldsymbol{T}_{m,2} \times \boldsymbol{T}_{m,3} \tag{3}$$

where

- $T_{m,1}$ number of ways the erasures can be distributed over the columns they occupy;
- $T_{m,2}$ number of ways of distributing the occupied columns over the matrix columns while preserving the relative order of the occupied columns;
- $T_{m,3}$ number of distinct permutations of the occupied columns. If all the occupied columns have the same size, $T_{m,3} = 1$.

Let the number of occupied columns be n and the number of erasures in column k be C_k. The first two terms can then be written as

$$\boldsymbol{T}_{m,1} = \prod_{k=1}^{n} \binom{M}{C_k} \tag{4}$$

$$\boldsymbol{T}_{m,2} = \binom{N}{n}. \tag{5}$$

$T_{m,3}$ depends on how many *different sizes* of occupied columns the basic pattern has. Let the number of different sizes be s and let z_j be the number of columns of size number j. (Note that $\sum_{j=1}^{s} z_j = n$). Then

$$\boldsymbol{T}_{m,3} = \prod_{j=1}^{s} \binom{n - x_{j-1}}{z_j} \tag{6}$$

where $x_j = x_{j-1} + z_j$ and $x_0 = 0$.

Now we state the following important result.

Result 1: For any number of erasures i, the generated patterns from all basic patterns are distinct and exhaustive, i.e., they span all possible i-erasure patterns.

The proof of the above result is straightforward. Result 1 implies that

$$\sum_{m=1}^{f_i} G_m = \binom{MN}{i}. \tag{7}$$

We next show that a great deal about the recoverability of the generated patterns can be deduced from the recoverability of their mother basic pattern.

IV. Recoverability Study

The basic patterns can be classified to recoverable basic patterns (RBPs) and unrecoverable basic patterns (UBPs). A basic pattern is recoverable if and only if its erasures are confined to the first column and first row of the matrix. For any number of erasures i in an $M \times N$ matrix, the number of RBPs, g_i, can be found as

$$g_i = \begin{cases} i, & \text{if } i \le \min(M,N) \\ \min(M,N), & \text{if } \min(M,N) < i \le \max(M,N) \\ M+N-i, & \text{if } \max(M,N) > i < M+N \\ 0, & \text{if } i \ge M+N. \end{cases} \tag{8}$$

The relation between an RBP and its generated patterns, from the recoverability point of view, is given in the following result.

Result 2: If the basic pattern is recoverable, then all the patterns it generates are recoverable.

The proof follows from the following argument. In any pattern generated from an RBP, all columns, except possibly the first column, contain a single erasure and therefore can be recovered by column-wise decoding. The row-wise decoding would then recover all erasures in the first column, if any.

Let the number of patterns generated from all RBPs be V_i. Then

$$V_i = \sum_{m=1}^{g_i} G_m \tag{9}$$

and, according to Result 2, all of them are recoverable. In obtaining (9), it is assumed that the numbering of the basic patterns is done such that the first g_i patterns of the f_i basic patterns are the recoverable ones.

The converse of Result 2 is not true; when a basic pattern is unrecoverable, the patterns it generates may or may not be recoverable. In order to identify the recoverable patterns generated from unrecoverable basic patterns, we need to find all the UBPs and then to find, for each of the UBP, the recoverable patterns it generates.

To simplify the process of enumerating the UBPs, let us partition the UBP to a *body* and two branches called *arm* and *leg* as demonstrated in Fig. 2(c). We can find all the UBPs for a given number of erasures i by implementing the following algorithm.

1) Construct all the possible bodies for this number of erasures.
2) For each body, the first UBP is constructed by placing all the remaining erasures (if any) on the right of the body as an arm. If the arm is full, the remaining erasures are placed in the leg.
3) The following UBPs for the same body will be generated by moving an erasure at a time from the arm and place it in the leg until no more erasures remain in the arm or the leg is completely filled.

Let the number of UBPs of size i be h_i. For each of these patterns, we need to find the recoverable patterns it generates. Let F_m be the number of recoverable patterns generated from the mth UBP, $m = 1, 2, \ldots, h_i$. As explained in Section III, there are three different movements for the erasures of a basic pattern to produce all its generated patterns; the first type of movement (distribution over columns) is the only one that decides the recoverability of the generated pattern.

The first type of movement involves the distribution of the arm erasures and the nonarm (body and leg) erasures over the columns they occupy. Since there is one erasure per column for the arm, the arm erasures can always be recovered by column-wise decoding regardless of their position in the column. Therefore, the recoverability is only determined by the distribution of the nonarm erasures over the columns. Let a be the number of erasures in the arm. Then the term $T_{m,1}$ may be written as the product of two terms

$$T_{m,1} = M^a \times Y_m. \tag{10}$$

The first term calculates all possible distribution of the a arm erasures, each in the column it occupies, whereas the second term calculates all the possible distribution of the nonarm erasures. The subpatterns indicated in Y_m are the ones that determine the recoverability of the overall pattern. Let the number of the recoverable subpatterns from the set Y_m be S_m. Then the number of recoverable patterns generated from the mth UBP is given by

$$F_m = M^a \times S_m \times T_{m,2} \times T_{m,3} \tag{11}$$

where $T_{m,2}$ and $T_{m,3}$ are as defined in (5) and (6). The term S_m can be obtained algorithmically. While it was possible to evaluate S_m exactly for some values of i, it was obtained only as an upper bound for other values of i. The number, W_i, of *definitely* recoverable patterns generated from all the h_i UBPs of size i is then given by

$$W_i = \sum_{m=g_i+1}^{g_i+h_i=f_i} F_m. \tag{12}$$

The total number of i-erasure recoverable patterns, R_i, is given by the sum of V_i and W_i . The number of unrecoverable error patterns is then upper bounded by $U_i \le \binom{MN}{i} - R_i$.

A little more can be added to the tightness of the bound on P by considering the number of unrecoverable erasures in an unrecoverable pattern after decoding. The bound in (2) assumes that when a pattern is unrecoverable all its erasures are unrecoverable. In reality, some erasures in an unrecoverable pattern may be recovered. In any decoding iteration, any erasure that is found alone in a column or a row can be recovered. Therefore, all arm erasures can be recovered. Also, many other erasures

TABLE I
NUMERICAL RESULTS OF THE NOVEL APPROACH WHEN APPLIED TO A 16 × 16 SPC PRODUCT MATRIX

Number of erasures (i)	Total no. of Patterns $\binom{MN}{i}$	Number of RBPs (g_i)	Number of UBPs (h_i)	Total basic Patterns (f_i)	% $\frac{V_i}{\binom{MN}{i}}$	% $\frac{W_i}{\binom{MN}{i}}$	% $\frac{R_i}{\binom{MN}{i}}$	% $\frac{e_i}{i}$
4	1.7E+08	4	1	5	~99%	~1%	~100%	100%
5	8.8E+09	5	2	7	~95%	~5%	~100%	80%
6	3.7E+11	6	5	11	~88%	~12%	~100%	67%
7	1.3E+13	7	8	15	~76%	~24%	~100%	57%
8	4.1E+14	8	14	22	60%	39%	99%	51%
9	1.1E+16	9	21	30	44%	55%	99%	47%
10	2.8E+17	10	32	42	29%	69%	98%	46%
11	6.2E+18	11	45	56	17%	80%	97%	48%
12	1.3E+20	12	65	77	9%	86%	95%	52%
13	2.4E+21	13	88	101	4%	88%	92%	56%
14	4.1E+22	14	121	135	1%	87%	88%	60%
15	6.7E+23	15	160	175	0%	82%	83%	63%
16	1.0E+25	16	213	229	0.12%	76%	76%	66%
17	1.4E+26	15	278	293	0.03%	67%	67%	68%
18	1.9E+27	14	367	381	0.01%	58%	58%	70%

in the body and the leg can be recovered. A simple algorithm was set up, utilizing the concepts defined in this work (arm, leg, bodies, and others), to find the average number of unrecoverable erasures in an unrecoverable pattern.

V. RESULTS AND DISCUSSION

The analysis of Section III was applied to an SPC product code of size16×16. The results are tabulated in Table I. It is clearly observed that the number of basic patterns is extremely small compared to the total number of patterns. Recalling that the recoverable patterns are solely determined from the basic patterns, one realizes the great simplification provided by our approach.

Column 8 of Table I shows the ratio of the recoverable patterns R_i to the total number of i-erasure patterns. The high percentages obtained prove the high capability of the SPC product code in recovering erasures beyond its minimum-distance measure (2). This capability was only partially explored in [10] for $i = 4, 5$. The full exploration of the code capability is credited to the novel approach developed here. Moreover, the figures in Column 6 of the table proves that V_i constitutes the large percentage of the total recoverable patterns (particularly for small i which are the dominant terms in P). Therefore, a bound that is sufficiently tight may be obtained by considering V_i only.

The last column of Table I gives the ratio of the number of erasures after decoding to the number of erasers before decoding, for the unrecoverable patterns. While none of the erasures in the four-erasure unrecoverable patterns are recoverable, 20% –52% of the erasures are recoverable for larger patterns.

Substituting the results of the above analysis in (1) produces an upper bound on the post decoding erasure rate P. The bound is sketched in Fig. 3 (Curve E) for a 16×16 matrix. Also shown on the same figure are some simulation points. The closeness of the simulation points to the curve verifies the extreme tightness of the bound.

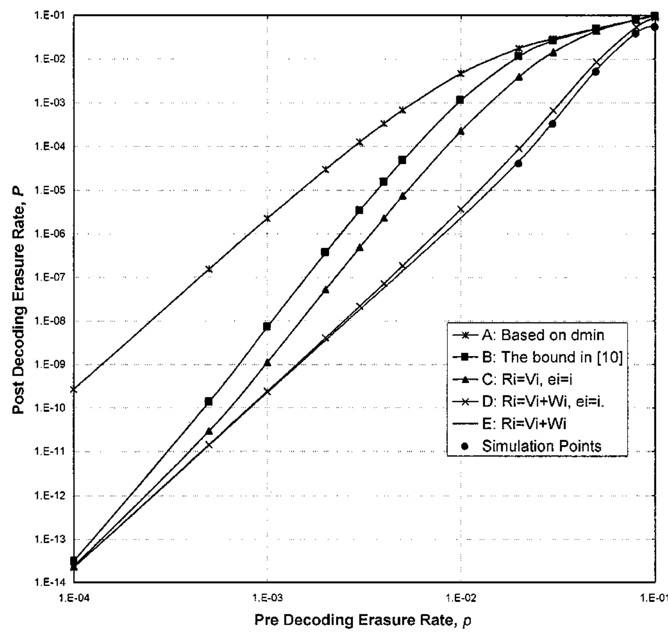

Fig. 3. Various upper bounds on the post-decoding erasure rate of a 16×16 SPC product code.

Other curves are plotted in Fig. 3. They are meant to show the explicit contribution of each tightening factor derived in this paper. The topmost curve (Curve A) refers to the case where the code can only recover patterns that are within the capability set by its minimum distance. In our notation, it corresponds to the case $R_i = 0$ and $e_i = i$ for $i > 3$. The next curve (Curve B) corresponds to the bound derived in [10]. It is equivalent to the case $R_i = 0$ for $i > 5$ and $e_i = i$, $i > 3$. The curve below (Curve C) refers to the case $R_i = V_i$ and $e_i = i$. Here we are assuming that the only recoverable pattern are those patterns generated from the recoverable basic patterns. Finally, the curve next to lowest (Curve D) refers to the case $R_i = V_i + W_i$ and $e_i = i$. The following conclusions are evident.

1) Basing the recoverability of the SPC product code on its minimum distance criterion (Curve A) is pessimistic. The results are far from accurate.
2) Including the exact evaluation of U_4 and U_5 only (corresponding to the bound in [10]) improves the estimation of the code performance (Curve B), but is still far from the actual performance.
3) By estimating only the recoverable patterns generated from the RBPs and assuming everything else is unrecoverable (Curve C), the bound is tightened significantly. This may be attractive as a first estimate of the code capability of erasure decoding as it is fairly easy and straightforward to calculate V_i as compared to calculating W_i.
4) Using e_i instead of i adds very little to the tightness of the bound. (Compare Curves D and E).

VI. Conclusion

A novel analytical approach to study the structure of the erasure patterns in SPC product codes was developed. The approach helps in finding the unrecoverable patterns to a very accurate degree. The results of this approach lead to an extremely tight upper bound on the post decoding erasure rate.

Acknowledgment

The author is thankful to H. Selmi for his constructive ideas and fruitful discussions.

References

[1] G. Caire, G. Taricco, and G. Battail, "Weight distribution and performance of the iterated product of single-parity-check codes," in *IEEE GLOBECOM*, 1994, pp. 206–211.

[2] E. Biglieri and V. Volski, "Approximately Gaussian weight distribution of the iterated product of single-parity-check codes," *Electron. Lett.*, vol. 30, no. 12, pp. 923–924, June 1994.

[3] L. Ping, S. Chan, and K. Yeung, "Efficient soft-in-soft-out sub-optimal decoding rule for single parity check codes," *Electron. Lett.*, vol. 33, no. 19, pp. 1614–1616, Sep. 1997.

[4] ——, "Iterative decoding of multi-dimensional concatenated single parity check codes," in *IEEE Int. Conf. Communications*, vol. 1, 1998, pp. 131–135.

[5] X. Huang, N. Phamdo, and L. Ping, "BER bounds on parallel concatenated single parity check arrays and zigzag codes," in *Global Telecommunications Conf.—GLOBECOM*, 1999, pp. 2436–2440.

[6] H. Ohta and T. Kitami, "A cell loss recovery method using FEC in ATM networks," *IEEE J. Select. Areas Commun.*, vol. 9, pp. 1471–1483, Dec. 1991.

[7] H. T. Lim and J. S. Song, "Cell loss recovery method in B-ISDN/ATM networks," *Electron. Lett.*, vol. 31, no. 11, pp. 849–851, May 1995.

[8] H. T. Lim, D. Nyang, and J. S. Song, "Improving the performance of cell loss recovery in ATM networks," *Electron. Lett.*, vol. 32, no. 17, pp. 1540–1542, August 1996.

[9] M. A. Kousa, A. K. Elhakeem, and H. Yang, "Performance of ATM networks under hybrid ARQ/FEC error control scheme," *IEEE/ACM Trans. Networking*, vol. 17, pp. 917–925, Dec. 1999.

[10] M. A. Kousa and A. H. Mugaibel, "Cell loss recovery using two-dimensional erasure correction for ATM networks," in *Seventh Int. Conf. on Telecommunication Systems*, Mar. 1999, pp. 85–89.