IJARCCE
ISSN (O) 2278-1021, ISSN (P) 2319-5940
International Journal of Advanced Research in Computer and Communication Engineering
Vol. 10, Issue 8, August 2021
DOI: 10.17148/IJARCCE.2021.10826

An Iterative Formation Method of Error Patterns Library Used in Product Codes Decoding Based on Syndrome-Normal

Xunhuan Ren [1], Konopelko Valery Konstantinovich [2], Tsviatkou Victor Yurievich [3]

PG Student, Dept. of Infor-Communication Technologies, BSUIR, Minsk, Belarus [1]

Professor, Dept. of Infor-Communication Technologies, BSUIR, Minsk, Belarus [2]

Associate Professor, Dept. of Infor-Communication Technologies, BSUIR, Minsk, Belarus [3]

Abstract: Product codes are usually applied in high data rate wireless communication systems to achieve good performance. The product code composed by the simple block code can be decoded by a decoding method based on the syndrome-norm. Syndrome-norm decoding method uses lookup table to roughly determine the type of the error pattern and then take specific corresponding decoding method. The scale of the lookup table will increase with the rise of the bits of the maximum errors since it requires to store all the possibilities of the error types. However, the existing pattern library formation method is high computational complexity and fail to form a pattern library with error bits above six. This paper proposed a mathematical model for fast generating a library based on the iterative expansion of the error patterns, which makes it possible to shorten the computational complexity in comparison with the known forming approaches and support higher error correction capability.

Keywords: product code decoding, iterative formation, syndrome, norm, syndromic-norm decoding, library of error patterns.

I. INTRODUCTION

Error-correcting and error-detecting codes play a vital role in applications used for networking, satellite communication, compact disks, and many other purposes. Product code is a technique to achieve a larger code with higher error-correction code capabilities by using simple block codes as component codes in the constructions. In encoding process of the product code, sequence code is first transformed into a code matrix, and then the row and column check bit are calculated [1-3]. The encoding process for the product code based on hamming code is shown in Fig.1.

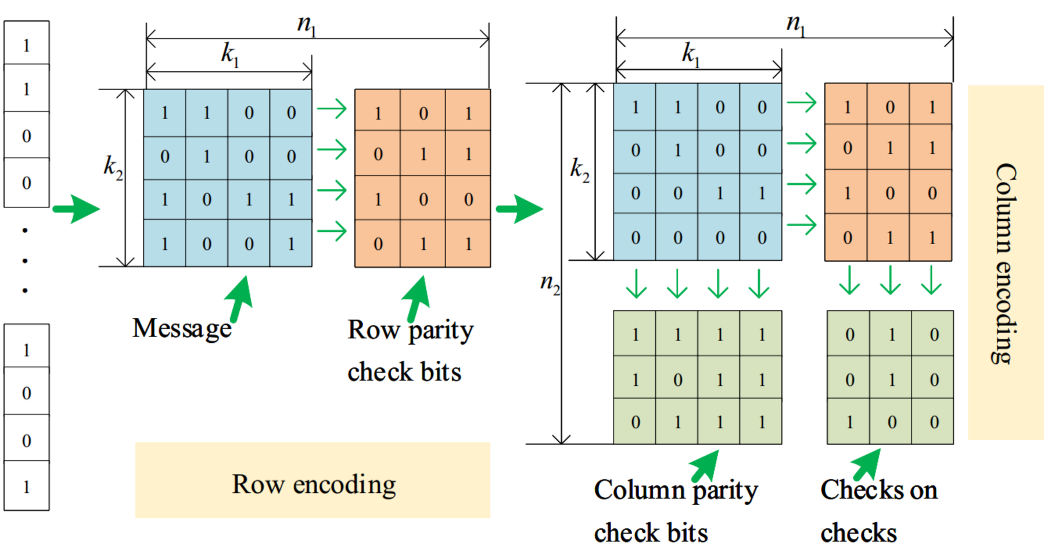

Fig. 1 Encoding process for the Hamming (7,4) product code

Currently, the mainstream decoding methods for product code composed by block code is based on iterative method, which adopts multi-stages to detect and correct corresponding errors, such as [4-10]. In each stage, the traditional syndrome-based decoding process conducted along the row or the column, respectively. Another one decoding method, which named

IJARCCE
ISSN (O) 2278-1021, ISSN (P) 2319-5940
International Journal of Advanced Research in Computer and Communication Engineering
Vol. 10, Issue 8, August 2021
DOI: 10.17148/IJARCCE.2021.10826

as Syndrome-norm decoding method [11], is less well-known. This method is based on the lookup table for decoding the product code. This method can fast match the target pattern from the pattern library by using the norm, which is calculated based on the result of the syndrome vector, can uniquely determine the base pattern and the corresponding set of error patterns and then applied a predefine correction method to correct the errors. The library of error patterns consists of some basic patterns that delegate a set of error patterns with a common characteristic. If one pattern can be transformed into another pattern by exchanging internal two-row or two-column, then we consider that these two patterns share the same characteristics. The decoding ability of this method is wholly dependent on the error pattern library. Therefore, one of the limitations of this method is that the formation of the matching library is very time-consuming, so it is tough to form a pattern with a lot of errors.

In the recent ten years, two library formation methods have been proposed to solve the problem mentioned above. To simplify the problem, all the patterns are referred as matrices, and in each matrix, each error position is marked as one. Then, it assumed that the maximum number of errors in a certain pattern is not above the square root of the total number of elements of it. A formation method has been proposed in [12]. This method first enumerates all the possibilities of error patterns in a given size. This pattern set is named as original error pattern set. Then it compares all these patterns to selects some patterns with distinct feature vector to adding to the library. The feature vector used in this method including the rank of the matrix. Another formation method is based on sub-classified, which proposed in [13]. It tends to first category the original patterns set into several subset according to some criterion then conduct the comparing them in a smaller set rather than directly compare them with each other. This method saves more time when comparing with the rank method. However, the sub-classified method still needs to generate all possible error patterns as the original pattern, which may cost a lot of time. As a result, it is necessary to develop a new library formation method that is not needed to list all the error patterns. In this paper, an iterative formation method of error patterns library used in product codes decoding based on Syndrome-Normal has been proposed.

II. RELATED METHODS

A. Syndrome-Norm Decoding for Product codes

In the scheme of syndrome-norm decoding of product codes [11], the syndromes and norms of rows and columns are first calculated. Then, these two parameters can match with a unique error pattern in the library. But before match operation, it is useful to shorten the search area by calculating the rough number of the error. After the matching operation, we will correct the error by a specified corrected method. The diagram of the whole decoding process is presented in Fig. 2.

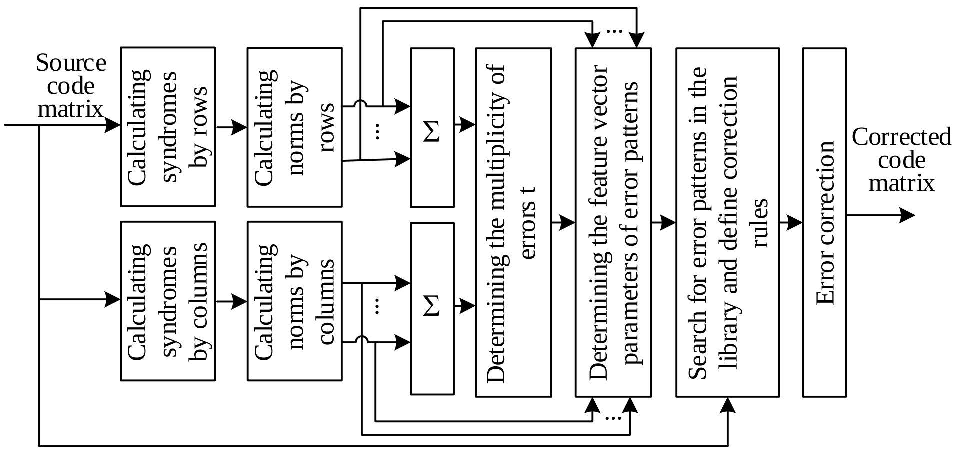

Fig. 2 Diagram of the whole decoding process

B. Pattern Library and Existing Formation Methods.

The pattern library applied in the Syndrome-Norm can be presented as a set of matrix $\{M_R(n)\}_{(n=\overline{1,N(t)})}$. Each element in this set should satisfy the following equations.

$$M_R(n) = \left\| m_R(n,i,j) \right\|_{(i=\overline{1,t},\, j=\overline{1,t})}, \quad (1)$$

$$\sum_{i=1}^{t}\sum_{j=1}^{t} m_R(n,i,j) = t\ , \quad (2)$$

$$\neg\exists n_1 \neg\exists n_2 \left(M_R(n_1)\right) = f_{RT}\left(f_{CT}\left(M_R(n_2), j_c\right), i_r\right) \text{, when } j_c = \overline{1,T},\ i_r = \overline{1,T} \quad (3)$$

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 10, Issue 8, August 2021
DOI: 10.17148/IJARCCE.2021.10826

Where $N(t)$ – the number of errors (see in table 1); f_{RT} – function that exchange two rows; f_{CT} – function that exchange two column. T – the number of the total exchanges, $T = C_t^2$.

TABLE I TOTAL NUMBER OF ERROR PATTERNS UNDER DIFFERENT GIVEN NUMBER OF ERROR BIT

Total Number of Error Patterns	**Number of error bit *t***						
	2	3	4	5	6	7	8
$N(t)$	3	6	16	34	90	212	558
$N_R(t)$	6	84	1820	53130	1947792	85900584	4426165368
$N_o(t)$	3	28	455	10626	324632	12271512	553270671
$N_d(t)$	2	14	52	210	620	2150	6498

In [12], a library formation method has been proposed. its mathematic model is presented in the following.

$$\{M_R(n)\}_{(n=\overline{1,N})} = f_M\left(\{M_T(n_r), p_{Rank}(n_r), p_{RE}(n_r), p_{CE}(n_r), p_{Ws}(n_r)\}_{(n_r=\overline{1,N_R(t)})}\right), \quad (4)$$

Where $M_T(n_r)$ – one initial matrix (error pattern), it should satisfy the formulation (2), $N_R(t) = t^2!/(t^2-t)!t!$ – the total number of the original matrix. f_M – function that used to select the distinct type of error patterns from the set $\{M_T(n_r)\}_{(n_r=\overline{1,N_R(t)})}$; $\{p_{Rank}(n_r), p_{RE}(n_r), p_{CE}(n_r), p_{Ws}(n_r)\}$ – the parameter that identify the different type of error pattern. p_{Rank} – the rank of the matrix $M_T(n_r)$; p_{CE} – sum of each column in $M_T(n_r)$; p_{RE} – sum of each row in $M_T(n_r)$; p_{Ws} – vector of the weighted sums of elements of rows and columns of the $M_T(n_r)$. These parameters can be calculated according to the following formulation.

$$p_{RE}(n_r) = \sum_{i=1}^{t} 10^i s_R(n_r, i), \quad (5)$$

Where $\{s_R(n_r,i)\}_{i=\overline{1,t}} = f_{sort}\left(\{v(n_r,i)\}_{(i=\overline{1,t})}\right)$; $v(n_r,i) = \sum_{j=1}^{t} m_T(n_r,i,j)$

$$p_{CE}(n_r) = \sum_{j=1}^{t} 10^j s_C(n_r, j), \quad (6)$$

Where $\{s_C(n_r,j)\}_{j=\overline{1,t}} = f_{sort}\left(\{v(n_r,j)\}_{(j=\overline{1,t})}\right)$; $v(n_r,j) = \sum_{i=1}^{t} m_T(n_r,i,j)$

$$p_{Ws}(n_r) = f_{sort}\left(\{p_w(n_r,k)\}_{(k=\overline{1,ij})}\right), \quad (7)$$

$$p_w(n_r,k) = \left(10\sum_{x=1}^{t} m_T(n_r,x,j) + \sum_{y=1}^{t} m_T(n_r,i,y)\right) m_T(n_r,i,j) \quad , \quad k = \overline{1,(j-1)+(i-1)\cdot t}$$

In order to accelerate the process of generating the library of error pattern, a library forming method based on sub-classified has proposed in [13]. The core mathematic model of this method is shown in the following.

$$\{M_R(n)\}_{(n=\overline{1,N(t)})} = f_M\left(\{M_o(n_o), p_{RE}(n_o), p_{CE}(n_o), p_{RZ}(n_o)\}_{n_o=\overline{1,N_o(t)}}\right), \quad (8)$$

$$M_o(n_o) = f_{MD}\left(\{M_T(n_r)\}_{(n_r=\overline{1,N_R(t)})}\right) \text{ when } n_r = \overline{1,N_R(t)},\ N_o(t) = (t^2-1)!/(t^2-t)!(t-1)!, \quad (9)$$

Where $M_o(n_o)$ – subset of error patterns; f_{MD} – preprocessing function. $p_{RZ}(n_o)$ – the number of the row that every element is zero, who can calculate by the following formula.

$$p_{RZ}(n_o) = \sum_{i_1=1}^{t} r(n_o, i) \quad (10)$$

Where $r(n_o,i) = \begin{cases} 1, \sum_{j=1}^{t} m_o(n_o,i,j) = 0, \\ 0, \sum_{j=1}^{t} m_o(n_o,i,j) \neq 0. \end{cases}$

IJARCCE

ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Vol. 10, Issue 8, August 2021

DOI: 10.17148/IJARCCE.2021.10826

III. PROPOSED METHODS

In order to further decrease the time-consuming, we have proposed an iterative error pattern library formation, whose mathematical model is presented in the following.

$$\{M_R(n,t)\}_{n=\overline{1,N(t)}} = \{M_x(n_x,t)\}_{n_x=\overline{1,N_X(t)}} \cup \{M_x(n_x,t)^T\}_{n_x(t)=\overline{1,N_X(t)}} \quad (11)$$

$$\{M_x(n_x,t)\}_{n_x=\overline{1,N_X(t)}} = f_M\left(\{M_d(n_d,t), p_{Rank}(n_d,t), P_{RC}(n_d,t), P_{Ws}(n_d,t)\}_{n_d=\overline{1,N_d(t)}}\right),$$

$$\{M_d(n_d,t)\}_{n_d=\overline{1,N_d(t)}} = f_{add}\left(\{M_x(n_x,t-1)\}_{n_x=\overline{1,N_X(t-1)}}\right),\ N_d(t)=N_X(t-1)\cdot(t^2-t+1),$$

$$M_x(n_x,1)=m_x(n_x,1)=1 \text{ when } N_x(1)=1,$$

$$\neg\exists n_1 \neg\exists n_2 \left(M_R(n_1)=\left(M_R(n_2)^T\right)\right) \text{ when } n_1=\overline{1,N(t)},\ n_2=\overline{1,N(t)}, \quad (12)$$

Where $M_x(n_x,t)$ – the base error pattern under t-th iteration, $M_x(n_x,t)=\|m_x(n_x,t,i,j)\|_{(i=\overline{1,t},\, j=\overline{1,t})}$; $M_d(n_d,t)$ – extended error pattern; f_{add} – function that add one another error to the existing base error pattern.

$$P_{RC}=f_{sort}(p_{RE},p_{CE}). \quad (13)$$

Based on the formula of (11) – (13), the following algorithm is proposed for expanding error patterns and adding one error to the extended matrix.

Algorithm 1 Extending and Adding one more error

```
Input: set of basis error pattern on t − 1-th iteration {M_x^(t−1))}
Output: set of basis error pattern on t-th iteration {M_d^(t)}
{M_x^(t))} ← ∅,{M_mid} ← ∅
// extending matrix
for (i ← 1; i ≤ n; i++)) do
    [Row, Column] ← size(M_x^(t−1)(i))
    M_mid(i) ← zeros(Row + 1, Column + 1)
    for (r ← 1; r ≤ Row + 1; r++)) do
        for (c ← 1; c ≤ Column + 1; c++)) do
            if r ≤ t − 1 || c ≤ t − 1 then
                M_mid(i)(r, c) ← M_x^(t−1)(i)(r, c)
            else
                M_mid(i)(r, c) ← 0
            end if
        end for
    end for
end for
// Adding one error to each elements
Index ← 1
for each element M in {M_mid} do
    for (r ← 1; r ≤ Row + 1; r++)) do
        for (c ← 1; c ≤ Column + 1; c++)) do
            if M(r, c) = 0 then
                P ← M
                P(r, c) ← 1
                M_x^(t)(Index) ← P
                Index++
            end if
        end for
    end for
end for
return {M_d^(t)}
```

Fig. 3. shows the block diagram of the proposed method, which can be seen that a new set of error patterns can be obtained based on the former set of error patterns, go through a series of expanding operations, and selecting operations. By contrast with other generating methods, the search range of representative error patterns of the proposed is less, which may reduce the time for creating the whole library.

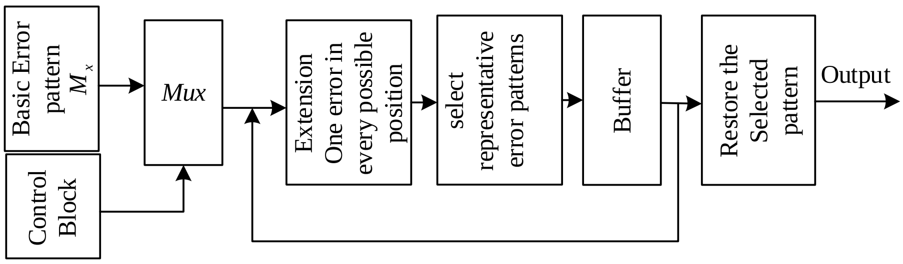

Fig. 3 Block diagram of error pattern generation method based on the iterative extension

IV. EXPERIMENT AND RESULTS

The comparison of the computational speed is conducted on the basis of the implementation of three different formation methods. The platform of the experiment environment in Matlab under the windows 10 operating system. The average execution time for generating the error patterns library used by the different methods is summarized in table 2.

TABLE 2 AVERAGE TIME FOR GENERATING ERROR PATTERN LIBRARIES FOR DIFFERENT METHOD

The time consumed by the different methods	**The number of Error *t***						
	2	3	4	5	6	7	8
Method in [12]	<1 s	<1 s	<1 s	7 s	10 m	42 m	>80 h
Method in [13]	<1 s	<1 s	<1 s	5 s	8 m	36 m	>80 h
Proposed Method	<1 s	<1 s	<1 s	<1 s	<1 s	2 s	8 s

We can see from the table. 2 that when the number of errors is below 5, all three methods can generate a corresponding library within one second, however, when the number of errors is surpassed 5, there are emerge some differences. For the first method, it will take 7 seconds to erect an error pattern library when t is 5, and it will take 10 minutes to create another library when t is 6. The second method costs lower time by comparing with the rank method, but it also needs 8 minutes to form a library when t is 6. The proposed method has the best result. For all different numbers of t, the time spent by the proposed method to construct the same libraries that build by the former two methods is all within one second. It is convinced that the proposed method has a better performance in terms of processing speed when compare with the first method and second method.

V. CONCLUSION

A model for forming a library of error patterns based on iterative expansion of the error patterns for the syndrome-norm decoding of iterative codes are proposed in this paper. The proposed method differs from the known rank method and subclassified method by it adopts an iterative forming method to build a library of error patterns. By iterative adding extra one error to the patterns that generated in the former iteration and eliminating redundant error patterns and retaining those representative patterns that have distinct feature vectors. The feature vectors are obtained according to the formula (13). The experiments has proved that the proposed method is a very efficient method, which requires less time to build a same library that formed by the rank method and by the subclassified method because the proposed method cancel to enumerate all the permutation of the possible error situation. The result of the proposed method can be applied in the syndrome-norm method, which is a decoding method based on the library matching.

REFERENCES

[1] S. Lin and D. J. Costello, Error Control Coding, 2nd edition.,Prentice Hall, Upper Saddle River, NJ, USA, 2004.

[2] P. Elias, “Error-free Coding,” in Transactions of the IRE Professional Group on Information Theory, vol. 4, no. 4, pp. 29–37, Sept. 1954.

[3] F. Chiaraluce and R. Garello, “Extended Hamming product codes analytical performance evaluation for low error rate applications,” IEEE Transactions on Wireless Communications, vol. 3, no. 6, pp. 2353–2361, Nov. 2004.

[4] Pyndiah, Ramesh, et al, “Near optimum decoding of product codes,” IEEE GLOBECOM. Communications: The Global Bridge. IEEE, 1994.

[5] R. M. Pyndiah, “Near-optimum decoding of product codes:block turbo codes,” IEEE Transactions on Communications, vol. 46, no. 8, pp. 1003–1010, Aug. 1998.

[6] Fu B and Ampadu P, “On hamming product codes with type-II hybrid ARQ for on-chip interconnects,” IEEE Trans Circuits Syst I, Reg Papers, vol. 56, no. 9, pp. 2042–2054, Sept. 2009.

[7] L. Ping, S. Chan, and K. L. Yeung, “Iterative decoding of multi-dimensional concatenated single parity check codes,” in Proc. ICC ’98, vol. 1, pp. 131–135, 1998.

ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Vol. 10, Issue 8, August 2021

DOI: 10.17148/IJARCCE.2021.10826

[8] M. Maheswari, G. Seetharaman, "Hamming product code based multiple bit error correction coding scheme using Keyboard Scan Based Decoding for on Chip Interconnects Links," Applied Mechanics and Materials, vol. 241-244, pp. 2457–2461,Dec. 2012.

[9] Chen, Y. H, Truong, T. K, et al, "A lookup table decoding of systematic (47, 24, 11) quadratic residue code," Information sciences, pp. 2470–2477, 2009.

[10] Yang C, Emre Y, Chakrabarti C, "Product code schemes for error correction in MLC NAND flash memories," IEEE Transactions on Very Large Scale Integration (VLSI) Systems , pp.2302–2314, 2011.

[11] Lipnitski V.A, Konopelko V.K, "The theory of syndrome norms in the permutation decoding action unjammable codes," Doklady BSUIR, vol. 1, no. 1, pp. 146–157, 2000.

[12] Konopelko V.K, Lipnitski A.V and Spichekova N. V, "Point pattern classification and classical number partition problem, " Doklady BSUIR, vol. 51, no. 5, pp. 112–117, 2010.

[13] Smolyakova O.G., Konopelko V.K, "Classification vectors of errors at two-dimensional coding of the information," in conference MRTI-BSUIR, 2009, p. 71–72.

BIOGRAPHY

Xunhuan Ren – PG student of department of info communication technologies of Belarusian state university of informatics and radioelectronics.

Konopelko Valery Konstantinovich – Doctor of Engineering, professor of department of info communications of Belarusian state university of informatics and radioelectronics.

Tsviatkou Victor Yurievich – Doctor of Engineering, associate professor, head of department of info communications of Belarusian state university of informatics and radioelectronics.