DE GRUYTER OPEN

Journal of ELECTRICAL ENGINEERING, VOL 68 (2017), NO4, 322–324

COMMUNICATIONS

Run length limited error control codes construction based on one control matrix property

Peter Farkaš, [*] **Frank Schindler** [**]

In this manuscript a simple method is presented for constructing run length limited error control codes from linear binary block codes. The run length limited properties are obtained via addition of a carefully chosen fixed binary vector - a modifier to all codewords without introducing any additional redundancy. Modifier selection is based on a specific property, which can be found in some of the linear binary block codes control matrices. Similar known methods are based on properties of generator matrices. However some codes are specified via control matrices, for example low density parity check codes. The method proposed in this letter could be applied to some of them directly. This is illustrated in this manuscript using example in which a run length limited low density parity check code is obtained from Gallager code.

K e y w o r d s: run length limited error control codes, control matrix, linear block codes, modifier

1 Introduction

A run-length limited error control code (RLL-ECC) has constrained run-lengths of identical symbols and can be used for error control. The research of RLL-ECC can be traced back to the late eighties of the 20-th century [1–2]. In [3] a new modified construction was presented yielding improved RLL properties for DC free error-correcting codes. In [4] trellis RLL-ECCs were constructed. In [5] a RLL-ECC for one error correction was proposed. In [6] a systematic approach was given for obtaining RLL-ECCs from binary error-correcting codes. The storage system industry motivated research into relatively complex RLL-ECCs based on turbo codes and low density parity check (LDPC) codes [7–11]. Recently, new RLL Codes were proposed for visible light communication (VLC) systems together with soft decoding methods [12–14]. In [15] algorithms were presented for code design using finite state machines which simultaneously provide a coding gain while also mitigating flicker in VLC. In [16–23] RLL-ECCs were constructed using the arrangement illustrated in Fig. 1, from linear binary block codes (LBBC) using modifiers obtained based on generator matrix properties. This method has a disadvantage, that it cannot be used directly for codes that are specified via control matrices, for example LDPC codes. In contrast to this method, the approach proposed in this letter could be applied to some of them directly because the modifiers are selected using a specific property, which can be found in some LBBC control matrices.

2 Basic properties of Linear Binary Block Codes Matrices

If C is an k-dimensional LBBC in which each codeword $\mathbf{c}$ has length n, it could be defined using a $[k \times n]$ generator matrix over a finite field $GF(2)$. The control matrix for C is defined in [24] as

$$C = \left\{\mathbf{c};\ \mathbf{c}\mathbf{H}^{\top} = \mathbf{0}\right\}. \tag{1}$$

Each row of $\mathbf{H}$ describes one control equation, which each codeword must fulfil. For example the following $\mathbf{H}$ matrix

$$\mathbf{H} = \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}, \tag{2}$$

which describes a $[7,4,3]$ Hamming code, defines for each codeword $\mathbf{c} = (c_6, c_5, c_4, c_3, c_2, c_1, c_0)$ following 3 control equations

$$c_3 + c_2 + c_1 + c_0 = 0\,, \tag{3}$$

$$c_6 + c_5 + c_1 + c_0 = 0\,, \tag{4}$$

$$c_6 + c_4 + c_2 + c_0 = 0\,. \tag{5}$$

N o t e . Each LBBC contains one all zero codeword and therefore an infinite run of zeros could occur in an infinite sequence of codewords.

3 Description of the proposed construction method

In practice it is often necessary to avoid long runs of identical symbols in encoded sequences. The practical

* Faculty of Electrical Engineering and Information Technology, Slovak University of Technology, Ilkovičova 3 812 19 Bratislava, Slovakia, p.farkas@ieee.org, ** Faculty of Informatics, Pan-European University, Tematínska 10, 851 05 Bratislava, Slovakia

DOI: 10.1515/jee-2017–0046, Print (till 2015) ISSN 1335-3632, On-line ISSN 1339-309X © 2017FEI STU

reason to do it could be for example that synchronisation is not well supported by such long runs. One method how it could be done is to use a modifier $\mathbf{m}$ in cascade with the LBBC encoder and add it to each codeword. On the receiving end this operation could be reversed by adding the same vector to each codeword before it is input to the decoder as illustrated in Fig. 1.

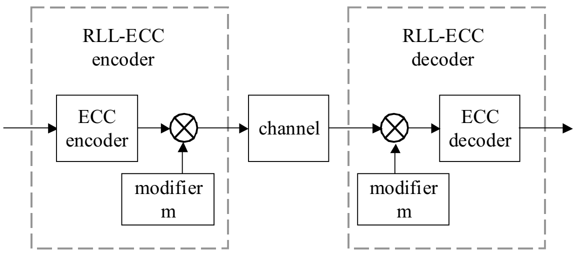

Fig. 1. Illustration of the basic principle for obtaining RLL-ECC using an LBBC encoder and decoder in cascade with additions of modifiers

In [16–23] the modifier formation was done using properties of $\mathbf{G}$ matrices described and proven in [7]. Unfortunately, the LBBC (*eg* LDPC codes) are often defined via their matrix and we cannot use this approach directly. Therefore in the following we will describe a different property which can be found sometimes in an $\mathbf{H}$ matrix and which then allows us to specify the modifier based on it.

PROPERTY. *If the control equation of a linear binary block code C contains a set E with an even number of symbols, then inverting an odd number of symbols from E in all the codewords of C creates a new code C' in which no codeword has symbols which are all equal to zero or all equal to one in E. (C' is a coset code of C).*

P r o o f . Suppose that the codeword $\mathbf{c}$ from C is transformed by inverting some symbols from E (which we will denote $\mathbf{c}_E$) to a codeword $\mathbf{c}'$ from C' in such a way that all symbols corresponding to E in $\mathbf{c}'_E$ are the same. Let w be the even weight and n_E be the even length of $\mathbf{c}_E$ respectively. To obtain $\mathbf{c}'_E$ from $\mathbf{c}_E$ either the w 1's or the $(n_E - w)$ 0's must be inverted. But since both w and n_E are even, this transformation would require an even number of inversions.

The application of the property could be illustrated using $[7, 4, 3]$ Hamming code. For example (3) contains four symbols c_3, c_2, c_1, c_0. Therefore we can select any modifier with seven bits which contains an odd number of ones in the four rightmost coordinates *eg* $\mathbf{m} = (0, 0, 0, 0, 0, 1, 0)$.

Often it is possible to find in a control matrix more than one control equation with an even number of summands for some codes. In case the sets of summands in different equations are disjunctive, it is possible to apply an odd number of negations in each codeword coordinates subset corresponding to each such equation. Let us illustrate it using the control matrix (6) of the LDPC code, which was constructed by Gallager [25]. One can see that the first five rows specify five control equations with disjunctive sets of symbols, each containing four symbols. Therefore it is possible to negate an odd number of codeword coordinates in each corresponding subset using a modifier

$$\mathbf{H} = \begin{bmatrix}
1&1&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&1&1&1&1&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&1&1&1&1&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&1&1&1&1&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&1&1&1\\
1&0&0&0&1&0&0&0&1&0&0&0&1&0&0&0&0&0&0&0\\
0&1&0&0&0&1&0&0&0&1&0&0&0&0&0&0&1&0&0&0\\
0&0&1&0&0&0&1&0&0&0&0&0&0&1&0&0&0&1&0&0\\
0&0&0&1&0&0&0&0&0&0&1&0&0&0&1&0&0&0&1&0\\
0&0&0&0&0&0&0&1&0&0&0&1&0&0&0&1&0&0&0&1\\
1&0&0&0&0&1&0&0&0&0&0&1&0&0&0&0&0&1&0&0\\
0&1&0&0&0&0&1&0&0&0&1&0&0&0&0&1&0&0&0&0\\
0&0&1&0&0&0&0&1&0&0&0&0&1&0&0&0&0&0&1&0\\
0&0&0&1&0&0&0&0&1&0&0&0&0&1&0&0&1&0&0&0\\
0&0&0&0&1&0&0&0&0&1&0&0&0&0&1&0&0&0&0&1
\end{bmatrix}. \quad (6)$$

For example, we can use the following modifier

$$\mathbf{m} = (1\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 1\ 0\ 0\ 0). \quad (7)$$

Using LDPC code defined by (6) and modifier (7) in the arrangement illustrated in Fig. 1, we can get an RLL-LDPC code with a maximal run length equal to six. It is because in each quadruple of coordinates following each other in encoded sequence not all symbols are equal. Consequently, in the worst case there are at most six consecutive equal symbols inside each codeword and also in the sequence consisting of concatenation of such codewords.

5 Conclusion

In this short communication we proposed a method for constructing RLL-ECC based on one property, which could be found in some ECC control matrices. We illustrated it using a Hamming code and an LDPC code. The advantages of the method are as follows. It is very simple, any additional redundancy is introduced by it into the encoded sequence and decoding the underlying code does not have to be changed. The disadvantage is that in some cases the necessary property is not present in the particular control matrix and in such cases it cannot be applied.

Acknowledgements

This work was supported by the European Space Agency (Contract No: 4000117400/16/NL/NDe) and the Grant Agency Academic Alliance (grant GA/2016/10).

References

[1] M. A. Herro and L. Hu, "Error-Correcting Line Codes", *23rd Ann. Allerton Conf. Commun.*, Control, and Comp., Monticello, IL, October 1985, pp. 450–451.

[2] R. H. Deng and M. A. Herro, "DC-Free Coset Codes", *IEEE Trans. on Inform. Theory*, 1998, vol. 34, no. 4, pp. 786–792.

[3] A. Popplewell and J. J. O'Reilly, "Spectral Characteristics of a Class of DC free Error-Correcting Transmission Codes", *Electronics Letters*, 1988, vol. 24, no. 15, pp. 919–920.

[4] L. Yinyi and J. K. Wolf, "Combined ECC/RLL codes", *IEEE Trans. on Magn.*, 1988, vol. 24, no. 6, pp. 2527–2529.

[5] P. J. K. Lee and J. K. Wolf, "A General Error-Correcting Code Construction for Run-Length Limited Binary Channels", *IEEE Trans. on Inform. Theory*, 1989, vol. 35, no. 6, pp. 1330–1335.

[6] A. Popplewell and J. J. O'Reilly, "Runlength Limited Binary Error Control Codes", *IEE Proceedings I – Commun., Speech and Vision,* 1992, vol. 139, no. 3, pp. 349–355.

[7] L. Jun and L. Jaejin, "Error Correcting RLL Codes using High Rate RSC or Turbo Code", *Electronics Letters*, 2001, vol. 37, no. 17, pp. 1074–1075.

[8] B. Vasic and K. Pedagani, "Run-Length-Limited Low-Density Parity Check Codes based on Deliberate Error Insertion", *IEEE Trans. on Magn.*, 2004, vol. 40, no. 3, pp. 1738–1743.

[9] Zongwang Li and B. V. K. V. Kumar, "Low-Density Parity-Check Codes with Run Length Limited (RLL) Constraints", *IEEE Trans. on Magn.*, 2006, vol. 42, no. 2, pp. 344–349.

[10] H. F. Chou, Y. L. Ueng, M. C. Lin and M. P. C. Fossorier, "An RLL-Constrained LDPC Coded Recording System Using Deliberate Flipping and Flipped-Bit Detection", *IEEE Trans. on Commun.*, 2012, vol. 60, no. 12, pp. 3587–3596.

[11] S. S. Arslan, J. Lee and T. Goker, "Cycle Slip Detection and Correction Through Classification of Modulation Code Failures", *IEEE Trans. on Magn.*, 2013, vol. 49, no. 9, pp. 4988–4998.

[12] H. Wang and S. Kim, "Soft-Input Soft-Output Run-Length Limited Decoding for Visible Light Communication", *IEEE Photonics Technol. Letters*, 2016, vol. 28, no. 3, pp. 225–228.

[13] H. Wang and S. Kim, "Bit-Level Soft Run-Length Limited Decoding Algorithm for Visible Light Communication", *IEEE Photonics Technol. Letters*, 2016, vol. 28, no. 3, pp. 237–240.

[14] Z. Babar *et al* "Unity-Rate Codes Maximize the Normalized Throughput of ON-OFF Keying Visible Light Communication", *IEEE Photonics Technol. Letters*, 2017, vol. 29, no. 3, pp. 291–294.

[15] C. E. Mejia, C. N. Georghiades, M. M. Abdallah and Y. Albadarneh, "Code Design for Flicker Mitigation Visible Light Communications Using Finite State Machines", *IEEE Trans. on Communications*, 2017, PP, (99), pp. 1–1.

[16] P. Farkaš and H. Weinrichter, "Transcontrol Codes with Run-Length Limitation", *AEU Int. J. Electron. Commun.*, 1996, vol. 50, no. 6, pp. 353–356.

[17] M. Šechny and P. Farkaš, "Some New Runlength-Limited Convolutional Codes," *IEEE Trans. on Commun.*, 1999, vol. 47, no.7, pp. 962–966.

[18] P. Farkaš, W. Pusch, M. Taferner and H. Weinrichter, *EU Int. J. Electron. Commun.* , 1999, vol. 53, no. 3 pp. 161–166.

[19] P. Farkaš, "Turbo-codes with RLL properties", *IEE Colloq. on Turbo Codes Digital Broadcasting – Could It Double Capacity?*, London, UK, November 1999, pp. 13/1-13/6.

[20] R. Chomist and P. Farkaš, "Some Extended Hamming Transcontrol Codes", *Radioelektronika*, 15-th Internat.Czech-Slovak Scient.Conf., Brno, Czech Republic, May 2005, pp. 342–345.

[21] R. Chomist and P. Farkaš, "Extended Golay Code with Transcontrol Properties", *6th Internat. Conf. on Dig. Sign. Processing and Multimedia Commun.*, September 2005, Košice, Slovakia, pp. 102–105.

[22] P. Farkaš and R. Chomist, "Reed Muller-Codes with Run Length Properties", *Joint IST Workshop on Mobile Future and the Symp. on Trends Commun.*, Bratislava, Slovakia, October, 2004, pp. 66–69.

[23] K. Farkašová *et al* "Construction of Error Control Run Length Limited Codes Exploiting some Parity Matrix Properties", *Journal of Electrical Engineering*, 2015, vol. 66, no. 3, pp. 182-184.

[24] J. I. Hall, "Notes on Coding Theory – Chapter 3, Linear Codes", *(Michigan State University*, Michigan, 2010) http://users.math.msu.edu/users/jhall/classes/codenotes/Linear.pdf, accessed March 2017.

[25] R. G. Gallager, "Low-Density Parity-Check Codes", *(MIT Press*, Cambridge, 1963).

Received 3 March 2017