ACSIJ Advances in Computer Science: an International Journal, Vol. 4, Issue 2, No.14 , March 2015
ISSN : 2322-5157
www.ACSIJ.org

ACSIJ
WWW.ACSIJ.ORG

Error Control Coding in Optical Fiber Communication Systems: An Overview

Majid Hatamian[1], Hamid Barati[1], Samaneh Berenjian[2] , Alireza Naghizadeh[3] and Behrooz Razeghi[4]

[1] Department of Computer Engineering, Dezful Branch, Islamic Azad University
Dezful, Iran
majid.hatamian.h@ieee.org
hbarati@iaud.ac.ir

[2] Department of Computer Engineering and Information Technology, Amirkabir University of Technology
Tehran, Iran
sama.scientist@aut.ac.ir

[3] Department of Computer Engineering, University of Guilan
Guilan, Iran
alireza.naghizadeh.a@ieee.org

[4] Department of Electrical Engineering, Ferdowsi University of Mashhad
Mashhad, Iran
behrooz.razeghi.r@ieee.org

Abstract

In sending data from one point to another, such as transferring data between various components of a computer system, due to the existence of electromagnetic waves and other issues such as noise and attenuation, information may be changed in the middle of the track. Therefore, it is critical for receiver to ensure the accuracy of the information. For this reason, error control coding plays an important role in communication channels. This is specially true for optical communication systems as one of the most important mediums used for data transmission. In this paper, we present methods of error control coding in optical fiber communication systems. For this purpose, we introduce two categories of error correction codes. The most important types of error control coding techniques in optical fiber communications are: Reed-Solomon (RS), Bose-Chaudhuri-Hocquenghem (BCH), Product, and Low Density Parity Check (LDPC). Furthermore, an in-depth analysis for these error correction codes has been performed.

Keywords: *Optical Fiber, Error Control Coding, Primitive Polynomial, Linear Block Codes, Convolutional Codes.*

1. Introduction

The basic of optical fiber is a very thin fiber that is sometimes made of plastic or most often of glass which is responsible to transmit the data. Numerous advantages such as high bandwidth, low attenuation, low weight and high speed, are among the factors that have led to the increasing use of this technology [1].

Optical fiber communication systems have the capacity to transmit large volumes of data at very high speed over thousands of kilometres. Optical fibers provide much greater bandwidth and offer low power loss compared to metal cables. They are also much thinner and lighter which make them easier to install. Along with these advantages, it is still possible to effectively increase transmission capacity and decrease the costs with error control coding [2,3].

Generally, error detection can be done in two main forms. In the first approach, when the receiver detects an error in a message, it automatically requests the sender to resend the message. This process is repeated until the message can be received without error or the error continues beyond a predetermined number of transmissions. This method is called Automatic Repeat Query (ARQ). In the second approach, designed redundancy allows receiver to detect and correct a limited number of errors occurring in the message without the need to request sender for additional repeat requests. The second method is called Forward Error Correction (FEC) [4,5,6].

Error control coding consists of error detection and correction procedures. If during these processes an error was detected, it can be corrected. The error control coding is one of the most important elements of every modern optical fiber communication system. In optical fibers, FEC systems are typically known as binary error correction codes. Error correction codes have been successfully used

in wireless and wired communications to offer an error free transmission with high spectral efficiency [7,8].

In this paper, four types of error control coding methods include: RS, BCH, Product and LDPC codes are introduced. The mathematical structure of them is demonstrated and coding/decoding concepts for each of them is explained.

The rest of this paper is organized as follows. Section 2 describes abnormalities in optical fiber channels. Section 3 explains types of error correction codes. Section 4 is about fundamental mathematical concepts of error control coding. Section 5 presents Reed-Solomon, BCH, Product, and LDPC codes as the most important techniques for error control coding in optical fibers. Finally, Section 6 concludes this paper.

2. Abnormalities in Optical Fiber Channels

2.1 Dispersion

Dispersion takes place when the pulses pass through the optical fiber. By overspreading these pulses, it limits the available bandwidth. Dispersion in optical fibers can be divided into two types, Intermodal (multipath) and Intramodal (chromatic). Intermodal dispersion is caused by different distance lengths of the modes in the fiber and different effective velocities which results in flattening of the transmitted pulses through the fiber. This type of dispersion occurs in multi-mode fibers and is known as modal dispersion. Intramodal dispersion is a term used to describe the spreading of a light pulse as it travels down a fiber when light pulses launch close together (high data rates). In this way, they spread too much and result in errors and loss of information. Intramodal dispersion occurs in single-mode fibers and causes the pulse broadening in these fibers [1].

2.2 Noise

Noise is an ever present part of all systems. Any receiver must confront with noise. In analog systems, noise spoils the quality of the received signal. But in digital communication systems, noise debilitates the throughput. The reason is that, noise requires retransmission of packets or redundant coding to recover the data in the presence of errors. Combination of electronic circuits, optical components such as add/drop multiplexers, optical cross-connects and fiber optics are factors that have led to the occurrence of noise. We have two types of noise: 1- External noise: noise whose sources are external such as man-made noise or industrial noise, atmospheric noises, etc. 2- Internal noise: noise which is discovered within the receiver or communication system such as shot noise, thermal noise (white noise), miscellaneous internal noise, etc [1].

2.3 Nonlinear Effects

In an optical fiber, light is restricted to a very small lateral sector. For this reason, even mild optical powers lead to high optical intensities. Since, light propagates over considerable distances in a fiber, nonlinear effects often have vital effects. This is especially true about fibers which are used to transmit short pulses. In fact, it can be argued that these effects are dependent on the intensity of light. Scattering and Kerr effects are instances of nonlinear effects. For example, Self-Phase Modulation (SPM) is one of the consequences of the Kerr effect. SPM occurs when a light wave in the fiber experiences a nonlinear phase delay which results from its own intensity. Cross-Phase Modulation (CPM) occurs when two different waves with two different wavelengths, propagate together in a fiber [1,5].

2.4 Attenuation

In optical fiber, attenuation is the reduction in intensity of the light beam with respect to distance it travels through a fiber. Attenuation also affects the propagation of waves and signals in optical fibers. It is defined as the ratio of the optical output power (after light propagates the distance d) to the input power (overall optical throughput of an optical fiber) and it can be written as [1],

$$P(\mathrm{o}) = \mathrm{P(i)}e^{-\alpha_t d} \tag{1}$$

where, α_t is the total attenuation coefficient and it can be obtained as follows:

$$\alpha_t\,(\mathrm{dB/\ km}) = \frac{10}{d} \times \log_{10}\left(\frac{P(\mathrm{i})}{P(\mathrm{o})}\right) \tag{2}$$

3. Types of Error Correction Codes

Error correction codes are generally divided into two categories: Linear Block and Convolutional Codes. In the following, we introduce these two types of error correction codes.

3.1 Linear Block Codes

These codes are called linear because linear combinations of codewords and the word itself belongs to linear block

codes. In linear block codes, in order to protect data against errors, information source data is divided in forms of blocks with length k symbols. If we assume that $U=(u_1,u_2,...,u_k)$ is one of these blocks, thus its associated codeword would be $V=(v_1,v_2,...,v_n)$,where $n\geq k$. The first k symbols of the codeword V is the data block itself. In other words:

$$v_1=u_1,v_2=u_2,....,v_k=u_k \quad (3)$$

The $n-k$ remaining symbols, are parity symbols of the code. In a way that the codeword V satisfies $H.V=O$. H is called the code's parity matrix and is defined by $H=[\mathrm{A}\,|\,I_{n-k}]$. A is a $n-k$ dimensional matrix that is permanently fixed and I_{n-k} is the identity matrix of degree $n-k$ [9,10].

3.2 Convolutional Codes

In convolutional codes, the initial data string is called U and the generated data string is V . For each k symbols of U , n symbols of V are generated. In convolutional codes, U and V represent strings of information blocks while in linear block codes, U and V represent blocks of information. That is in convolutional codes, U and V consist of k and n symbols block strings, respectively. The important thing is each n symbols block of V is dependent on the previous m blocks of the initial data in addition to the k symbols initial data at the same time. Convolutional codes are represented with $(\mathrm{n},\mathrm{k},\mathrm{m})$. Some of the most popular convolutional codes decoding algorithms are Fano, Viterbi and ZJ [11,12].

4. Fundamental Mathematical Concepts of Error Control Coding

The field F is a set of elements on which we can apply addition, subtraction, multiplication and division operations. A field with a limited number of elements is called a Finite Field or Galois Field (GF). In the fields, addition and multiplication have the characteristics of associative, distributive and commutative. When we talk about field we mean the number of elements it contains. Thus, one field is defined by {0,1} which is called binary field and is represented by $GF(2)$. But, $GF(2^m)$ includes all m bits combinations which we represent with different powers of α that are the primitive elements of *GF*. In this case, $GF(2^m)$ includes 2^m members as following [13]:

$$0,\alpha^0,\alpha^1,\alpha^2,...,\alpha^{2^m-2} \quad (4)$$

Each *GF* has a *Primitive Polynomial* of degree m , with α as its root. Generally, in digital communication systems which work with binary data, error correction codes are constructed by elements of binary field $GF(2)$ or generalized field $GF(2^m)$. Typically, non-binary codes such as Reed-Solomon consist of $GF(2^m)$ and binary codes such as BCH consist of $GF(2)$. For example, consider that we want to represent the members of $GF(2^4)$. We assume that the primitive polynomial of this *GF* is as following [13]:

$$x^4+x+1 \quad (5)$$

Therefore, the members of this *GF* are:

$$0,\alpha^0,\alpha^1,\alpha^2,...,\alpha^{14} \quad (6)$$

For representing the members of a *GF*, there are different methods. One of these methods is Standard Basis [13,14] representation. In this method, each position of a m bits vector, represents a power of α . Thus, the rightmost place relates to $\alpha^0=1$, the next place relates to α^1 and so on. As an example, description for representing the members of $GF(2^4)$ in binary form, is as follows.

Since $m=4$, therefore we have a 4 bits vector that each of its digits represent a power of α . According to the above definitions, we have:

$$\begin{aligned} &0000=0\\ &0001=\alpha^0=1\\ &0010=\alpha^1\\ &0100=\alpha^2\\ &1000=\alpha^3 \end{aligned} \quad (7)$$

In the next step, we must find the other members of *GF* from these five members. As it is mentioned, α is the root of the primitive polynomial, so if we put α as x in the primitive polynomial, the answer of the equation is zero, we have:

$$\alpha^4+\alpha+1 \quad (8)$$

We know that in *GF*, addition and subtraction are equivalent and both are equal to XOR, so we have:

$$\alpha^4 = -\alpha - 1 = \alpha + 1 = 0011 \quad (9)$$

Thus, the other members of the *GF* are determined as follows:

$$\begin{aligned}
&0000 = 0\\
&0001 = \alpha^0 = 1\\
&0010 = \alpha^1\\
&0100 = \alpha^2\\
&1000 = \alpha^3\\
&0011 = \alpha^4 = \alpha + 1\\
&0110 = \alpha^5 = \alpha(\alpha^4) = \alpha(\alpha + 1) = \alpha^2 + \alpha\\
&\vdots\\
&0000 = \alpha^{15} = \alpha^4 + \alpha = \alpha + \alpha + 1 = 1
\end{aligned} \quad (10)$$

5. Error Control Coding in Optical Fiber Communication Systems

If we assume T is the time it takes for k symbols to be transferred without coding, T/k is the time it takes for one symbol to be transferred. After coding k symbols in a n symbols codeword, n symbols are transferred in time T and therefore the period of symbol is T/n which is lower than T/k.

In every error detection and correction system, some excess information called redundant is sent along with the original information. We represent the system that converts the k bits message to a codeword with length n, by the pair (n,k). The width of each symbol after coding is reduced by the k/n factor, and also the required bandwidth for the transfer of symbols is increased by the factor n/k, called the Bandwidth Expansion Ratio. In addition, we call k/n the code rate and represent it by R_c. In case of optical fiber communication systems that work in very high data rates $(R_c > 0.8)$, selection of the coding method that results in low overhead is very important. If R is closer to 1, means that the bandwidth has been used in a more efficient way. If R is closer to 0, means that we have more redundancy [2,4,10]. The structure of an optical fiber communication system is shown by Fig. 1.

Fig. 1. Structure of an optical fiber communication system.

5.1 The Reed-Solomon Code

The Reed-Solomon (RS) code is a subset of cyclic codes which are a subset of linear block codes themselves. RS is one of the most widely used error correction codes in optical fiber communications. This code can correct $2/(n-k)$ errors. RS is based on GF mathematical structure and like other error correction codes, transforms an information packet with length k to a codeword with length n. But RS differentiates itself with other codes such as Hamming or Parity. In this code, the number k does not mean k bits, but means k symbols that each symbol is a m bits member of $GF(2^m)$. Therefore in RS code, a mk bits string will be transformed to a mn bits codeword. Which is defined as $RS(n,k)$. In the RS, the number n relates to m which is shown as follows (n is the length of the block) [14]:

$$n = 2^m - 1 \quad (11)$$

The RS corrects up to $2/(n-k)$ errors in a n symbols information packet. But capability of correcting $2/(n-k)$ errors does not mean correcting $2/(n-k)$ bits, it means $2/(n-k)$ symbols contain errors. A string of information including elements of GF can be represented by a polynomial of X. The coefficients of different powers of X are the members of GF themselves. Assume that the string B includes the symbols b_0, b_1, b_2 and b_3. This string can be represented by the following polynomial [2]:

$$B(X) = b_3X^3 + b_2X^2 + b_1X + b_0 \quad (12)$$

5.1.1 The RS Encoder

The encoding process is done with a *Generator Polynomial*. The generator polynomial has $n-k=2t$ tandem roots. These roots are:

$$\alpha^{m_0}, \alpha^{m_0+1}.\alpha^{m_0+2}, ..., \alpha^{m_0+2t-1} \quad (13)$$

where, m_0 can be any number. But it must be selected carefully, because a good selection can result in optimization of some decoding stages. In other words, the generator polynomial $g(X)$ is defined as follows:

$$g(X) = (X + \alpha^{m_0})(X + \alpha^{m_0+1})...(X + \alpha^{m_0+2t-1}) \quad (14)$$

For encoding a string i(X) with k symbols in RS method, we must take these steps [9,14]:

Step 1: i(X) will be shifted by $n-k$ symbols to the left. Which creates $n-k$ symbols with a value of zero in the right side of i(X). It can be done by multiplying i(X) by X^{n-k}. Fig. 2 shows this procedure.

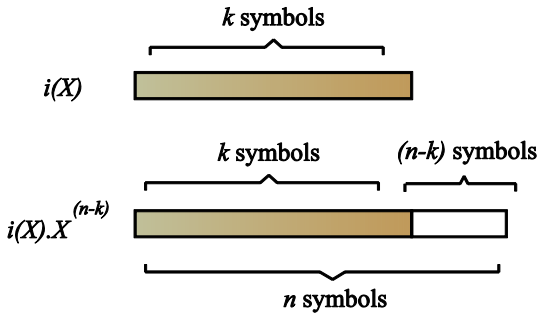

Fig. 2. Method of multiplying i(X) by X^{n-k}

Step 2: In this step, we must divide $i(X).X^{n-k}$ by the generator polynomial $g(X)$ and determine its remainder. The remainder is shown by $r(X)$.

Step 3: $r(X)$ will be placed in the $n-k$ empty places in the right side of $i(X).X^{n-k}$. In this way, a codeword is created that we call it $c(X)$ and is as follows:

$$c(X) = i(X)X^{n-1} + r(X) \tag{15}$$

To further clarify this process, we provide an example by $RS(15,11)$, we have:

$$\begin{aligned} &n = 15, k = 11 \\ &n-k = 2t = 4, m = 4 \end{aligned} \tag{16}$$

As previously mentioned, RS code can detect $(n-k)$ error symbols. It also can correct $(n-k)/2$ error symbols. In other words, RS (15,11) can detect m$(n-k)$ or 4(4)=16 error bits and correct m$(n-k)/2$ or 4(4)/2=8 error bits. Assume that we want to encode the following 44 bits string:

$$\begin{aligned} &i = (01111010010110111100011 \\ &00011110000100001000001) \end{aligned} \tag{17}$$

Then, i can be written as follows:

$$i = (\alpha^{10}, \alpha^{9}, \alpha^{8}, \alpha^{7}, \alpha^{6}, \alpha^{5}, \alpha^{4}, \alpha^{3}, \alpha^{2}, \alpha^{1}, 1) \tag{18}$$

We can write polynomial i(X) as follows:

$$\begin{aligned} i(X) = {} & \alpha^{10}\mathrm{X}^{10} + \alpha^{9}\mathrm{X}^{9} + \alpha^{8}\mathrm{X}^{8} + \alpha^{7}\mathrm{X}^{7} + \alpha^{6}\mathrm{X}^{6} \\ & + \alpha^{5}\mathrm{X}^{5} + \alpha^{4}\mathrm{X}^{4} + \alpha^{3}\mathrm{X}^{3} + \alpha^{2}\mathrm{X}^{2} + \alpha\ \mathrm{X}\ + 1 \end{aligned} \tag{19}$$

Step 1: In this step, the generator polynomial g(X), should be determined. So, we must select the roots of g(X). The roots of g(X) is selected as $\alpha^{12}, \alpha^{13}, \alpha^{14}$ and $\alpha^{15} = 1$ (how to determine α has been explained earlier). Because each of these four roots have one less α factor than the previous one and the calculation becomes easier. g(X) is created as below:

$$\begin{aligned} &g(X) = (X + \alpha^{12})(X + \alpha^{13})(X + \alpha^{14})(X + 1) \\ &g(\mathrm{X}) = X^{4} + \alpha^{9}X^{3} + \alpha^{13}X^{2} + \alpha^{6}X + \alpha^{9} \end{aligned} \tag{20}$$

Step 2: For encoding the string i(X), we have to divide $i(X).X^{4}$ by g(X) and determine the remainder:

$$\begin{aligned} &r(X) = i(X)X^{4} \bmod g(X) \\ &r(X) = \alpha^{7}X^{3} + \alpha^{7}X^{2} + \alpha^{8}X + 0 \end{aligned} \tag{21}$$

Step 3: We write c(X) as follows:

$$\begin{aligned} &c(X) = i(X)X^{4} + r(X) \\ &c(X) = (\alpha^{10}\mathrm{X}^{14} + \alpha^{9}\mathrm{X}^{13} + \alpha^{8}\mathrm{X}^{12} + \alpha^{7}\mathrm{X}^{11} + \alpha^{6}\mathrm{X}^{10} \\ &+ \alpha^{5}\mathrm{X}^{9} + \alpha^{4}\mathrm{X}^{8} + \alpha^{3}\mathrm{X}^{7} + \alpha^{2}\mathrm{X}^{6} + \alpha\mathrm{X}^{5} + \mathrm{X}^{4}) \\ &+ \alpha^{7}\mathrm{X}^{3} + \alpha^{7}\mathrm{X}^{2} + \alpha^{8}\mathrm{X} \end{aligned} \tag{22}$$

In this way, codeword is created as follows:

$$\begin{aligned} &(011110100101101111000110001110 \\ &000100001000011011101101010000) \end{aligned} \tag{23}$$

It can be seen, the length of c(X) is 60 bits. This means the initial information which was 44 bits, is transformed to 60 bits.

5.1.2 The RS Decoder

Decoding is the duty of the receiver. Immediately after receiving a codeword, the receiver detects whether an error occurred or not. The steps for error correction and detection in the RS coding method, are as below [2,3,14]:

1. Calculating the Syndrome Decoder.
2. Determining the Error Locator Polynomial (ELP) for calculating the error location in the codeword.
3. Determining the Error Evaluator Polynomial (EEP) for calculating the amount of errors in each location
4. Correcting the errors (if any).

In the following, the method for calculating syndrome decoder is explained. We mentioned that after taking the encoding steps, a polynomial appears which is divisible by $g(\mathrm{X})$, thus its remainder after the division by $g(\mathrm{X})$ is zero. In this way, if the data which is received by the receiver is divisible by $g(\mathrm{X})$, it indicates that no errors occurred. Otherwise, it is concluded that some errors have occurred and the codeword is invalid. It is also clarified that $g(\mathrm{X})$ can be written in form of Eq. (14). Also the roots of $g(\mathrm{X})$ are shown by Eq. (13).

In this case, if we call the codeword which is received by the receiver $v(\mathrm{X})$, the roots of $g(\mathrm{X})$ are also the roots of $v(\mathrm{X})$. So, to determine whether an error has occurred, it is sufficient to see whether $\alpha^{m_0}, \alpha^{m_0+1}, ..., \alpha^{m_0+2t-1}$ are the roots of $v(\mathrm{X})$ or not. For this purpose, just have to place these roots instead of the variable X in $v(\mathrm{X})$. If the value of v is zero with this placement, it indicates that no errors have occurred, otherwise we will know there is an error. So in general, for error detection, we must determine the value of $v(\mathrm{X})$ for each root of $g(\mathrm{X})$. The whole procedures which are explained so far is called syndrome calculation and can be summarized as:

$$\begin{aligned} S_0 &= v(\alpha^{m_0}) \\ S_1 &= v(\alpha^{m_0+1}) \\ &\vdots \\ S_{2t-1} &= v(\alpha^{m_0+2t-1}) \end{aligned} \qquad (24)$$

We call $S_0, S_1, ..., S_{2t-1}$ the syndromes of $v(\mathrm{X})$. If all the S's are zero, it means that we have no errors. Otherwise, there is an error and we must correct its location and value.

5.2 The Bose-Chaudhuri-Hocquenghem (BCH) Code

As it is mentioned before, since the BCH code acts on bits, so it is named binary. On the contrary, the RS code acts directly on symbols so it is called non-binary [15].

The BCH code is a random error correction code. This code is capable of correcting errors occurred randomly in the information string. The BCH and RS codes are also called random and burst error correction codes, respectively. The BCH code can also be called a generalized model of Hamming code that is capable of correcting multiple errors whereas the Hamming code is only capable of correcting one error. The BCH code similar to the RS code, is based on GF mathematical structure. In a BCH code, for positive integers $m \geq 3$ and $t < 2^{m-1}$, we have a BCH code with the following characteristics [12,15,16]:

$$n = 2^m - 1, n - k \leq m \times t, \mathrm{d}_{\min} \geq 2t + 1 \qquad (25)$$

where, $n = 2^{m-1}$ indicates the length of codeword (block), $n - k \leq m \times t$ indicates number of parity bits and $\mathrm{d}_{\min} \geq 2t + 1$ represents the minimum distance of code.

It should be obvious that this code can correct any combination of t errors in a block with length of $n = 2^{m-1}$. A code with this characteristic is called a BCH code with capability of correcting t errors. The generator polynomial for this code is determined by its roots which belong to $GF(2^m)$. If α is the primitive element of $GF(2^m)$, the generator polynomial $g(\mathrm{X})$ is a polynomial with the least degree and with coefficients of $GF(2)$ which its roots are as follows:

$$\alpha, \alpha^2, \alpha^3, ..., \alpha^{2t-1}, \alpha^{2t} \qquad (26)$$

It is important to understand that the most important concern about the BCH codes, is their difficult decoding [12,16,17].

5.3 Product Codes

Product codes can be called the first group of combinational codes which presented with the aim of more error correction capability. These codes are made by combination of two block codes (n_1, k_1, d_1) and

(n_2, k_2, d_2). This results are generated in a new block code with the form of $(n_1 n_2, k_1 k_2, d_1 d_2)$ [18,19].

5.3.1 Product Codes Encoder

A product code can be considered as a two-dimensional array. Its rows and columns are mapped by the first and the second codes, respectively. Fig. 3 shows how to form a two-dimensional array with elements of two block codes.

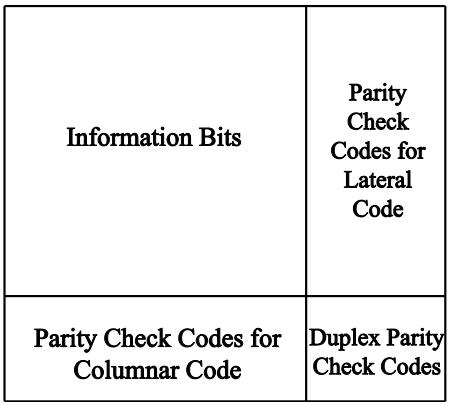

Fig. 3. Creation of the two-dimensional array in product codes.

A product code which is shown by C, is encoded using the first location of information bits in a $k_1 \times k_2$ matrix. Each column of the matrix $k_1 \times k_2$ is encoded by components (n_1, k_1, d_1) of code C_1. The result of encoding is stored in a $n_1 \times k_2$ matrix. Each row of the $n_1 \times k_2$ matrix is encoded by components (n_2, k_2, d_2) of code C_2 and the result of encoding is stored in a $n_1 \times n_2$ matrix. Therefore, we have a $(n_1 n_2, k_1 k_2)$ code.

The goal is to find the minimum distance of codeword C. We should find the smallest non-zero weight of codeword C. Assume that a codeword in its non-zero situation, for instance, in situation (i, j). In this case, the weight of column j is at least d_1. Thus, we have d_1 non-zero rows, including rows i. Each of these rows at least have weight d_2. It can be concluded that the weight of codeword C would be $d_1 \times d_2$. Fig. 4 demonstrates this problem. As we mentioned before, the smallest non-zero weight of codeword C should be found. The represented black squares depict the non-zero situations [19].

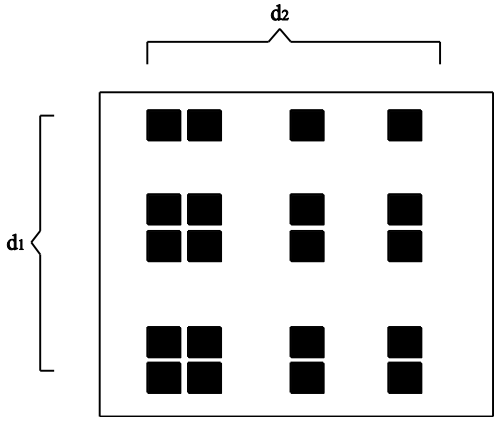

Fig. 4. How to find a non-zero minimum weight of codeword C

5.3.2 Product Codes Decoder

For decoding, first we need to decode all the columns by a decoder for C_1 and all the rows by a decoder for C_2, respectively. However, this method only ensures that errors with a weight of $\left\lfloor \frac{d_1 - 1}{2} \right\rfloor . \left\lfloor \frac{d_2 - 1}{2} \right\rfloor \approx \frac{d_1 d_2}{4}$ can be decoded. Fig. 5 shows how a product code can be decoded.

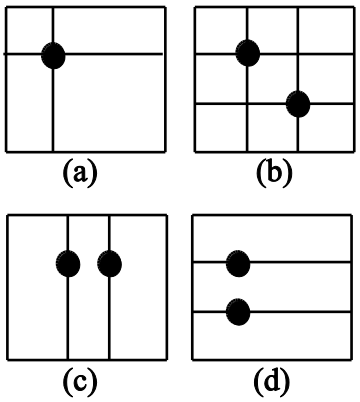

Fig. 5. Methods of finding errors in product codes for decoding. (a) occurrence of a single error, (b) occurrence of two errors in different rows and columns, (c) occurrence of two errors in the same rows and (d) occurrence of two errors in the same columns.

For example, we assume the codes C_1 and C_2 with components (3,2,2) and (5,4,2), respectively. Because the difference between n and k in these codes is 1, we conclude that each of the above codes has a redundancy bit. The product code resulting from the two above codes is (15,8,4).

5.4 Low Density Parity Check (LDPC) Codes

Low Density Parity Check (LDPC) codes are from the block codes family that their parity check matrix (H) has a sparse matrix form. A sparse matrix is a matrix that its non-zero elements are far fewer than its zero elements. Generally, LDPC codes are organized into two regular and irregular groups. In the regular group, the number of non-zero elements in all rows and all columns in the matrix H are equal. LDPC codes are one of the main rivals for Turbo codes. Both of them are used in current optical communications [20,21].

5.4.1 LDPC Codes Encoder

In this section, the creation of LDPC codes based on RS codes is explained. If we assume that α is the primitive element of $GF(\mathrm{q})$ such that $q = p^s$ is a power of a prime number, and also we pick the positive integer p such that $2 \le p < q$, the generator polynomial of code $RS(q-1, q-p+1, p-1)$ on $GF(\mathrm{q})$ is determined as follows [22]:

$$g(X) = (x-\alpha)(x-\alpha^2)...(x-\alpha^{p-2})$$
$$g(X) = g_0 + g_1 X + g_2 X^2 + ... + X^{p-2} \qquad (27)$$

In Eq. (27), $g_i \in GF(q)$. It is important to note that in Eq. (27), the polynomial has the lowest degree among other polynomials of the related code and all its $p-1$coefficients are non-zero. If we truncate this code by deleting $q-p-1$ information symbols, a reduced code $RS(q-1, 2, p-1)$ is created that only has two information symbols and the number of its codewords will be q^2. The generator matrix of such code has two rows and is written as below [23]:

$$G_b = \begin{bmatrix} g_0 & g_1 & g_2 & ... & 1 & 0 \\ 0 & g_0 & g_1 & g_2 & ... & 1 \end{bmatrix} \qquad (28)$$

5.4.2 LDPC Codes Decoder

LDPC codes are decoded in a repetitive manner. Decoding in LDPC codes is done using a graph named the Tanner graph. The edges in the graph, are those routes that information goes through. This graph is used to represent the matrix H [24].

A graph $G(\mathrm{V}, \mathrm{E})$ consists of the set of vertexes $V = \{v_1, v_2, ...\}$ and set of edges $E = \{e_1, e_2, ...\}$. Each edge e_k is introduced by the pair of vertexes (v_i, v_j) at its ends. Edges and vertexes are also called branches and nodes, respectively. The number of branches connected to each node is called the degree of the node and is shown by $d(v_i)$. For example, structure of the Tanner graph for the Hamming code (7,4) is shown by Fig. 6.

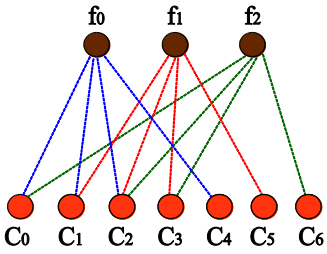

Fig. 6. Structure of the Tanner graph for the Hamming code (7,4).

Bit-flipping is a decoding algorithm for LDPC codes [24,25]. This method is based on the principle of restoration of syndrome equations by reversing some bits of the received string. The implementation steps of this algorithm are as follows:

Step 1: All v_i nodes (also called variable nodes) send their information to c_j nodes connected to them.

Step 2: c_j nodes (also call parity nodes), calculate the sum of the received bits and return the result (zero or one) to every node connected to them.

Step 3: v_i nodes, select a new value based on the majority vote by receiving this information from parity nodes connected to them and with the respect to their current value.

Step 4: The algorithm repeats from Step 1 until either all the syndrome equations are satisfied or the preset number of repeats is reached.

For example, assume that the received string on the receiver is $V = (11010101)$. The given matrix H for this graph is as follows:

$$\begin{matrix} 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 \end{matrix} \qquad (29)$$

Also, the Tanner graph for the given matrix H is shown by Fig. 7.

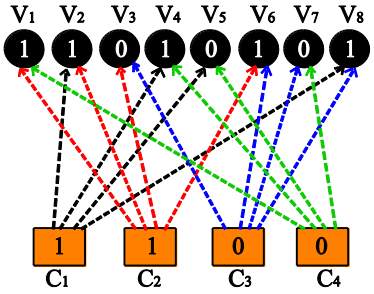

Fig. 7. Example of the Tanner graph with received string $V = (11010101)$ on the receiver.

Step 1: The information of string V is sent to the parity nodes and it can be seen that the values of the parity nodes after the information was sent, will be $C_1 = 0, C_2 = 0, C_3 = 1, C_4 = 1$.

Step 2 and 3:In these steps, we look at parity nodes with value 1. If the parity nodes with value 1 are connected to a variable node, the value of that variable node will become 0, by doing XOR the value of parity nodes connected to the variable node. According to this assumption, only the value of the variable node V_2 becomes 0. Because the two parity nodes C_1 and C_2 with value 1 are connected to V_2.

Step 4: Since the string 11010101 has been transformed to 10010101 and this new string is again sent to the parity nodes, and because all the parity equations are satisfied and $C_1 = C_2 = C_3 = C_4 = 0$, the decoding stops.

6. Conclusions

Error control coding is an important problem in modern communication systems. Since optical fibers are widely used for exchange of information all around the world, providing appropriate solutions for error control coding is crucial. In this paper, four types of error control coding methods which are used in optical fiber communication systems were introduced. RS is commonly used in most long haul optical fiber communication systems and is capable of correcting burst errors. BCH code is capable of correcting random errors which occur during transmission. Product codes have significantly less processing delay and are very useful in optical fiber. LDPC codes have good distance properties, particularly for long codeword lengths, therefore suitable in optical fiber communication systems. The structure of these methods were described and coding/decoding approaches were explained with a lot of practical example. Based on contents described so far, we present Table 1. In this table the advantages and disadvantages for most important criteria of each method is shown. As we can see, each method has its pros and cons. For example, RS and BCH have burst errors and random error correction capability. Product codes have less processing delay. LDPC codes suitable for long codeword lengths. They also support a fully parallelism decoding and this is very well when considering long codewords.

Table 1. Types, advantages and disadvantages of error control codes

Codes	Type	Advantages	Disadvantages
Reed-Solomon	Linear Block	Burst Errors correction and low computational complexity with predictable decoding capabilities	Alphabet size is as large as their length
BCH	Linear Block	Random Errors correction and ease of decoding by syndromes	Complex decoding
Product	Combinational	Decreased Bit Error Ratio performance and significantly less processing delay	Complex mathematical structure
LDPC	Linear Block	Good distance properties, particularly for long codeword lengths. They also support a fully parallelism decoding and this is very well when considering long code-words.	Complex encoding and inflexibility

References

[1] E. A. B. Saleh and M. C. Teich, Fundamentals of photonics, John Wiley and Sons Inc, 1991.
[2] B. P. Smith and F. R. Kschischang, “Future prospects for FEC in fiber-optic communications,” IEEE Journal of Selected Topics in Quantum Electron, vol. 16, no. 5, 2010, pp. 1245-1257.
[3] D.J. Costello, J. Hagenauer, H. Imai and S.B. Wicker, “Applications of error-control coding,” IEEE Transactions on Information Theory, vol. 44, no. 6, 1998, pp. 2531-2560.
[4] G. Gho, L. Klak and O. M. Kahn, “Rate-Adaptive coding for optical fiber transmission systems,” IEEE Journal of Lightwave Technology, vol. 29, no. 2, 2011, pp. 222-233.
[5] S. Cho, “Adaptive Forward Error Correction Scheme for

Real-Time Communication in Satellite IP Networks," KSII Transactions on Internet & Information Systems, vol. 4, no. 6, 2010, pp. 11-16.

[6] M. Zhang, Z. Wang, M. Guo, "A Method of Combining Scrambling Technology with Error Control Coding to Realize Both Confidentiality and Reliability in Wireless M2M Communication," KSII Transactions on Internet & Information Systems, vol. 6, no. 1, 2012, pp. 162-175.

[7] I. B. Djordjevic, M. Arabaci and L. L. Minkov, "Next generation FEC for high-capacity communication in optical transport networks," IEEE Journal of Lightwave Technology, vol. 27, no. 16, 2009, pp. 3518-3530.

[8] J. Climenta, V. Herranzb and C. Pereab, "Linear system modelization of concatenated block and convolutional codes," Elsevier Linear Algebra and its Applications Journal, vol. 429, no. 5–6, 2008, pp. 1191–1212.

[9] K. Fenga, L. Xua and F. J. Hickernellb, "Linear error-block codes," Elsevier Finite Fields and Their Applications Journal, vol. 12, no. 4, 2006, pp. 638–652.

[10] M. Magarini, R. J. Essiambre, B. E. Basch, A. Ashikhmin, G. Kramer and A. J. de Lind van Wijngaarden, "Concatenated coded modulation for optical communication systems," IEEE Photonics Technology Letters, vol. 22, no. 16, 2010, pp. 1244-1246.

[11] H. S. Mruthyunjayaa, G. Umeshb and M. Sathish Kumara, "Performance enhancement of optical fiber communication systems using convolution codes," International Symposium on Antenna Propagation and Communication ISAP2006, 2006, pp. 1-4.

[12] S. Lin and D. J. Costello, Error control coding: fundamentals and applications, Prentice Hall Publication, 2004.

[13] S. Choi, Y. Lee, H. Jeon and K. Kim, "Architecture of the high-speed standard basis multiplier with delay-boxes over $GF(2^m)$," IEEE International Conference on Electrical and Electronic Technology, 2001, pp. 339-402.

[14] D. Bleichenbachera, A. Kiayiasb and M. Yung, "Decoding interleaved Reed–Solomon codes over noisy channels," Elsevier Theoretical Computer Science Journal, vol. 379, no. 3, 2007, pp. 348–360.

[15] S. Y. Ho and D. J. Kleitman, "An odd kind of BCH code," Elsevier Discrete Applied Mathematics Journal, vol. 161, no. 9, 2013, pp. 1216–1220.

[16] F. Fu and S. Shen, "On the nonperiodic cyclic equivalence classes of Hamming codes and BCH codes," Elsevier Journal of Statistical Planning and Inference, vol. 92, no. 2, 2001, pp. 205-209.

[17] Z. Skupień, "BCH codes and distance multi- or fractional colorings in hypercubes asymptotically," Elsevier Journal of Discrete Mathematics, vol. 307, no. 7-8, pp. 990-1000, 2007.

[18] J. Justesen, "Performance of product codes and related structures with iterated decoding," IEEE Transactions on Communications, vol. 59, no. 2, 2011, pp. 407-415.

[19] A. P. T. Lau and J. M. Kahn, "Signal design and detection in presence of nonlinear phase noise," IEEE Journal of Lightwave Technology, vol. 25, no. 10, 2007, pp. 3008-3016.

[20] I. B. Djordjevic and B. Vasic, "Nonbinary LDPC codes for optical communication systems," IEEE Photonics Technology Letters, vol. 17, no. 10, 2005, pp. 2224-2226.

[21] Z. Zhang, V. Anantharam, M. J. Wainwright and B. Nikolic, "An efficient 10GBASE-T ethernet LDPC decoder design with low error floors," IEEE Journal of Solid-State Circuits, vol. 45, no. 4, 2010, pp. 843-855.

[22] B. Smith, M. Ardakani, W. Yu and F. R. Kschischang, "Design of irregular LDPC codes with optimized performance-complexity tradeoff," IEEE Transactions on Communications, vol. 58, no. 2, 2010, pp.489-499.

[23] A. Amraoui, A. Montanari, T. Richardson and R. Urbanke, "Finite-length scaling for iteratively decoded LDPC ensembles," IEEE Transactions on Information Theory, vol. 55, no. 2, 2009, pp. 473-498.

[24] J. M. R. Vaza and J. A. B. Geralda, "A fast LDPC encoder/decoder for small/medium codes," Elsevier Microelectronics Journal, vol. 44, no. 10, 2013, pp. 888-896.

[25] Q. Guoleia, "Interleaved Processing of Bit-Flipping Decoding For LDPC Codes," Elsevier Procedia Engineering Journal, vol. 15, 2011, pp. 1622–1625.

Majid Hatamian received his B.S. degree in Computer Hardware Engineering in 2013. He is currently working toward the M.S. degree in Computer Systems Architecture Engineering in Dezful Branch, Islamic Azad University, Iran. His major research experiences and interests include Wireless Networks and Mobile Communications, Cryptography and Data Protection, Security Issues in Wireless & Ad-hoc Networks and Machine Learning Techniques.

Hamid Barati received his B.S. degree in Computer Hardware Engineering, M.S. degree in Computer Systems Architecture Engineering and Ph.D. degree in Computer Systems Architecture Engineering in 2005, 2007 and 2014 respectively. Currently he is faculty of Islamic Azad University, Dezful Branch, Iran. His major research experiences and interests include Mobile Ad-Hoc Networks, Interconnection Networks & Energy-Efficient Routing and Security issues in Wireless Sensor Networks.

Samaneh Berenjian is an information technology engineer. Her research interests are in the areas of Intrusion Detection systems, Automated Intrusion Response Systems, Security protocols and Cryptographic algorithms where she is aiming to expand her research at security in e-health systems. She has worked in ISEC lab under the supervision of Dr. Mehdi Shajari since 2011. She has graduated with a M.Sc. in IT engineering from the Department of Computer Engineering and Information Technology, Amirkabir University of Technology.

Alireza Naghizadeh received his M.S. in Information Technology from University of Guilan, Iran in 2013. His current research interests are distributed and P2P networks, specializing in security, anonymity, game theory, network management and network architectures. He presents his papers in several academic seminars, workshops and has published over research papers in international journals and conferences.

Behrooz Razeghi was born in Mashhad, Iran, in 1989. He received the B.Sc. degree (First-class Honours) in Electrical and Communication Engineering from Sadjad University of Technology, Mashhad, Iran, in 2012. He is currently a research assistant in the Department of Electrical Engineering, Ferdowsi University of Mashhad, Mashhad, Iran. During his undergraduate studies, his research was focused on electronic circuits and

control systems. His current research interests fall into the broad areas of wireless communication, information theory, communication theory, computer network, information theoretic learning, and optimization methods in communications and networking. He is the author or co-author of more than 15 technical papers published in scientific journals and presented at international conferences. He has been a member of Technical Program Committees for several conferences and served as reviewer for numerous IEEE international journals and conferences.