Survey Paper of Binary Arithmetic Codes for Error Detection and Correction

Gautam Deo Verma[1], Prof. Suresh. S. Gawande[2], Prof. Satyarth Tiwari[3]

M. Tech. Scholar, Department of Electronics and Communication, Rkdf college of engineering, Bhabha University, Bhopal[1]

Guide, Department of Electronics and Communication, Rkdf college of engineering, Bhabha University, Bhopal[2]

Co-guide, Department of Electronics and Communication, Rkdf college of engineering, Bhabha University, Bhopal[3]

Abstract— In digital systems, the analog signals will change into digital sequence (in the form of bits). This sequence of bits is called as "Data stream". The change in position of single bit also leads to catastrophic (major) error in data output. Almost in all electronic devices, we find errors and we use error detection and correction techniques to get the exact or approximate output. Error detection and correction codes based on redundant residue number systems are powerful tools to control and correct arithmetic processing and data transmission errors. Decoding the magnitude and location of a multiple error is a complex computational problem: it requires verifying a huge number of different possible combinations of erroneous residual digit positions in the error localization stage.

Index Terms— Error Detection, Error Correction, Binary Arithmetic Codes

I. INTRODUCTION

Traditionally, channel coding is performed after source coding to protect the compressed bit stream sent over a noisy channel. For example, an image file is first compressed by using Discrete Cosine Transformation or arithmetic coding [1] with high coding efficiency. Then, the compressed sequence is further protected by Turbo code [2] or Hamming code [3] against channel noise. This traditional separate scheme lacks the cooperation between source and channel coding processes and may not result in the optimal performance. Recent studies have revealed that the joint operation of them leads to some advantages when compared with the traditional separately operated approach [4]. As certain implicit redundancy still exists in the bit streams when the encoder cannot ideally decorrelate the source symbols, it can be utilized in the joint scheme to improve the overall error correcting performance. Thus, it is possible for the joint scheme to outperform the separate approach [5].

Early works on joint source-channel coding were devoted to the study of error resilience in variable length codes (VLC). In particular, most of which were focused on the resynchronization ability of Huffman code [6]. The corresponding hard and soft decoding schemes based on maximum likelihood (ML) or MAP metrics are well-studied for a binary symmetric channel (BSC) with additive white Gaussian noise (AWGN). As arithmetic coding (AC) represents a source symbol using a fractional number of bits, it leads to a better compression efficiency and achieves the optimal entropy coding. However, the high compression ratio makes the codeword more sensitive to channel noise and is difficult to be resynchronized. Therefore, there is a growing interest in improving the robustness of AC against channel noise.

In [7], a forbidden symbol introduced by a reduction in the coding interval is adopted to detect the transmission error continuously. These errors can be detected when the forbidden region is visited. This continuous nature in error detection is exploited to improve the overall performance of the communication system [8]. It provides a tradeoff between the extra redundancy and the delay in detecting an error since its occurrence. Instead of the forbidden symbol, the insertion of markers in some particular positions of the input sequence plays the role of synchronization between the encoder and the decoder [9].

The markers which do not appear in the expected positions indicate transmission errors. Three strategies for the selection of the markers were studied in [10]. A better compression ratio can be achieved using an adaptive or an artificial marker scheme. The adaptive marker scheme selects the most frequent source symbol as the marker symbol while the artificial marker scheme creates an artificial marker with an arbitrary probability. Making use of the error detection capacity of AC, error correction is performed by sequential decoding, which successively removes the erroneous decoding paths. In [11], depth-first and breadth-first decoding algorithms were proposed with binary branching based on a null zone. The decoding paths are discarded due to the error detection capacity of the forbidden symbol. All the decoding paths with the lowest Hamming distance from the received sequence are preserved in a list.

In [12], a MAP criterion based on the context-based AC was proposed with the insertion of synchronization markers, where the symbol clock and the bit clock models were analyzed. The iterative decoding of error resilient AC concatenated with a convolutional code is adopted and its error correcting capability is validated with the transmission of images over an AWGN channel. A novel MAP decoding approach based on the forbidden symbol was proposed in [10], with a high

flexibility in adjusting the coding rate. Sequential decoding algorithms, such as stack algorithm and -algorithm, are adopted and the proposed system outperforms the separate approach based on convolutional codes in terms of error correcting capability. It is serially concatenated with channel codes and iterative decoding is employed to further improve the overall performance.

Thus, chaos control techniques can be adopted to further enhance the error correction performance. A sequential MAP estimation for CABAC coder was proposed in [13], which employs an improved sequential decoding technique to determine the tradeoff between complexity and efficiency. In [14], a look-ahead technique for AC decoder was proposed to allow quick error detection. Considering the improvement in the implementation efficiency, AC can be modeled as a finite-state machine corresponding to a variable-length trellis code. The trellis code based on AC was proposed in [2], where a list Viterbi decoding algorithm is applied on the corresponding trellis code and a cyclic redundancy check code is employed for detecting small Hamming-distance errors. The free distance of the corresponding AC-based VLC and its theoretical error correction performance were investigated in [29, 30]. Besides that, the practical implementations on this joint source-channel coding scheme were studied in [3] for high coding speed.

The error detecting capability of AC was analyzed in our previous paper [15]. Here we extend our previous work to tackle the problem of error correction in AC. An effective error correction technique utilizing the forbidden symbol is proposed, which predicts the occurrence of the subsequent forbidden symbols. With our approach, the forbidden region is theoretically expanded and so a better error correction performance is achieved. Furthermore, a generalized stack algorithm (SA) extending branches from the best node is also studied for the detection of the forbidden symbol beforehand. The MAP metric [10] is integrated with our approach to preserve the most probable decoding paths in the stack. The idea of our approach was briefly presented in [3], which mainly focuses on the forecasting of the forbidden symbols. Here, the procedures of AC with forecasted forbidden symbols are described in detail. More analyses and simulation results are provided to justify that the proposed scheme outperforms the look-ahead scheme [6] and the original MAP scheme [10] on the error correction performance, especially at a low coding rate.

II. LITERATURE REVIEW

In [1], Kristjane Koleci et al., depicts a productive execution of the iterative decoder that is the primary piece of the unscrambling stage in the LEDAcrypt cryptosystem, as of late proposed for post-quantum cryptography in light of low-thickness equality check (LDPC) codes. The execution we present endeavors the construction of the factors to speed up the disentangling system while keeping the region limited. Specifically, our attention is on the plan of an effective multiplier, the last option being a basic part likewise taking into account considering various upsides of the cryptosystem's boundaries, as it very well may be needed in later applications. We expect to give an engineering reasonable to minimal expense execution on both Field Programmable Gate Array (FPGA) and Application Specific Integrated Circuit (ASIC) executions. Concerning the FPGA, the complete execution time is 0.6 ms on the Artix-7 200 stage, utilizing all things considered 30% of the all out accessible memory, 15% of the absolute accessible Look-up Tables and 3% of the Flip-Flops. The ASIC blend has been performed for both STM FDSOI 28 nm and UMC CMOS 65 nm advancements. After rationale blend with the STM FDSOI 28 nm, the proposed decoder accomplishes a complete inactivity of 0.15 ms and a region control of 0.09 mm 2 . The post-Place&Route execution results for the UMC 65 nm show a complete execution season of 0.3 ms, with an area control of 0.42 mm 2 and a power utilization of at most 10.5 mW.

In [2], P. Santini et al. [2], iterative decoders utilized for unraveling low-thickness equality check (LDPC) and moderate-thickness equality check (MDPC) codes are not portrayed by a deterministic deciphering range and their blunder rate execution is generally surveyed through escalated Monte Carlo recreations. In any case, a few applications, similar to code-based cryptography, need ensured low upsides of the blunder rate, which are infeasible to evaluate through recreations, accordingly requiring the advancement of hypothetical models for the mistake pace of these codes. A few models of this sort as of now exist, however become computationally unmanageable for boundaries of commonsense interest. Different methodologies inexact the code gathering conduct through presumptions, which may not remain constant for a particular code. We propose a hypothetical examination of the blunder amendment ability of LDPC and MDPC codes that permits inferring tight limits on the mistake rate at the result of equal piece flipping decoders. Unique consideration is given to the situation of codes with little bigness. Single-cycle disentangling is researched through a thorough methodology, which doesn't need any presumption and results in a dependable blunder remedy capacity for any single code. We show an illustration of use of the new bound to the setting of code-based cryptography, where ensured mistake rates are expected to accomplish solid security levels.

In [3], J. Hu et al. [3], present a lightweight equipment plan for an as of late proposed quantum-safe key embodiment system in view of QC-LDPC codes called LEDAkem, which has been conceded as a cycle 2 contender to the NIST post-quantum normalization project. Existing executions center around fast while not many of them consider region or power productivity, which are especially unequivocal for minimal expense or power compelled IoT applications. The arrangement we propose targets augmenting the measurement of region proficiency by pivoting the QC-LDPC code portrayals among the square RAMs in digit level. In addition, upgraded parallelized processing strategies, languid gathering and square parcel are taken advantage of to work on key decapsulation as far as region and timing effectiveness. We show for example that our region streamlined execution for 128-digit security requires 6.82 x 105 cycles and 2.26 x 106 cycles to epitomize and decapsulate a common mystery,

individually. The region improved plan utilizes just 39 cuts (3 percent of the accessible rationale) and 809 cuts (39 percent of the accessible rationale) for key epitome and key decapsulation individually, on a little size low-end Xilinx Spartan-6 FPGA.

In [4], D. Zoni et al. [4], considering code-based cryptography, semi cyclic low-thickness equality check (QC-LDPC) codes are predicted as one of a handful of the answers for configuration post-quantum cryptosystems. The digit flipping calculation is at the center of the unraveling system of such codes when used to plan cryptosystems. A compelling plan should represent the computational intricacy of the deciphering and the code size needed to guarantee the security edge against assaults drove by quantum PCs. To this end, it is of foremost significance to convey productive and adaptable equipment executions to help quantum-safe public-key cryptosystems, since accessible programming arrangements can't adapt to the necessary exhibition. This original copy proposes a productive and adaptable engineering for the execution of the piece flipping methodology focusing on huge QC-LDPC codes for post-quantum cryptography. To exhibit the adequacy of our answer, we utilized the nine designs of the LEDAcrypt cryptosystem as delegate use cases for QC-LDPC codes appropriate for post-quantum cryptography. For every arrangement, our format engineering can convey an exhibition enhanced decoder execution for all the FPGAs of the Xilinx Artix-7 mid-range family. The test results exhibit that our upgraded design permits the execution of enormous QC-LDPC codes even on the littlest FPGA of the Xilinx Artix-7 family. Considering the execution of our decoder on the Xilinx Artix-7 200 FPGA, the test results show a normal exhibition speedup of multiple times across all the LEDAcrypt setups, contrasted with the authority enhanced programming execution of the decoder that utilizes the Intel AVX2 augmentation.

In [5], K. Koleci et al., this paper is based on cyclic redundancy check based encoding scheme. High throughput and high speed hardware for Golay code encoder and decoder could be useful in digital communication system. In this paper, a new algorithm has been proposed for CRC based encoding scheme, which devoid of any linear feedback shift registers (LFSR). In addition, efficient architectures have been proposed for both Golay encoder and decoder, which outperform the existing architectures in terms of speed and throughput. The proposed architecture implemented in virtex-4 Xilinx power estimator. Although the CRC encoder and decoder is intuitive and easy to implement, and to reduce the huge hardware complexity required. The proposed method it improve the transmission system performance level. In this architecture our work is to design a Golay code based on encoder and decoder architecture using CRC generation technique. This technique is used to reduce the circuit complexity for data transmission and reception process.

In [6], D. Zoni et al., Memories that operate in harsh environments, like for example space, suffer a significant number of errors. The error correction codes (ECCs) are routinely used to ensure that those errors do not cause data corruption. However, ECCs introduces overheads both in terms of memory bits and decoding time that limit speed. In particular, this is an issue for applications that require strong error correction capabilities. A number of recent works have proposed advanced ECCs, such as orthogonal Latin squares or difference set codes that can be decoded with relatively low delay. The price paid for the low decoding time is that in most cases, the codes are not optimal in terms of memory overhead and require more parity check bits. On the other hand, codes like the (24,12) Golay code that minimize the number of parity check bits have a more complex decoding. A compromise solution has been recently explored for Bose–Chaudhuri–Hocquenghem codes.

In [7], M. Baldi et al., this brief lays out cyclic redundancy check-based encoding scheme and presents an efficient implementation of the encoding algorithm in field programmable gate array (FPGA) prototype for both the binary Golay code (*G*23) and extended binary Golay code (*G*24). High speed with low-latency architecture has been designed and implemented in Virtex-4 FPGA for Golay encoder without incorporating linear feedback shift register. This brief also presents an optimized and low-complexity decoding architecture for extended binary Golay code (24, 12, 8) based on an incomplete maximum likelihood decoding scheme. The proposed architecture for decoder occupies less area and has lower latency than some of the recent work published in this area. The encoder module runs at 238.575 MHz, while the proposed architecture for decoder has an operating clock frequency of 195.028 MHz. The proposed hardware modules may be a good candidate for forward error correction in communication link, which demands a high-speed system.

In [8], Shivani Tambatkar et al., authors focus on Universal asynchronous receiver transmitter (UART) with BIST capability using different LFSR techniques and compared these techniques for the logic utilization in SPARTAN3 XC3S200-4FT256 FPGA devices. Work concluded by the comparison of LFSR techniques on the basis of hardware. Number of configurable logic blocks used after the implementation of the BIST techniques is increased from74 to 103 slices of the total slices. Area overhead results an increase in delay from 44.7 ns to 74.24 ns. The area overhead is somehow reasonable considering test performance obtained from these methods & gives the choice of the different LFSR methods with minimum area overhead or delay.

III. BINARY CODES

Block codes are referred to as (n, k) codes. A block of k information bits are coded to become a block of n bits. n=k + r, where r is the number of parity bits and k is the number of information bits.

The more commonly employed Block codes are:

1. Single Parity-Check Bit Code
2. Repeated Codes

3. Hadamard Code
4. Hamming Code
5. Convolution Code Codes
6. Cyclic Codes
7. Golay Code
8. Extended Golay Codes

Marcel Golay was born in Neuchatel, Switzerland in 1902. He was a successful mathematician and information theorist who was better known for his contribution to real world applications of mathematics than any theoretical work he may have done [9, 10]. Golay's sought the perfect code. Perfect codes are considered the best codes and are of much interest to mathematicians. They play an important role in coding theory for theoretical and practical reasons. The following is a definition of a perfect code:

A code C consisting of N codewords of length N containing letters from an alphabet of length q, where the minimum distance d=2e+1 is said to be perfect if:

$$\sum_{i=0}^{e}\binom{n}{i}(q-1)^{i}=\frac{q^{n}}{N} \qquad (1)$$

There are two closely related binary Golay codes. The extended binary Golay code G24 encodes 12 bits of data in a 24-bit word in such a way that any 3-bit errors can be corrected or any 7-bit errors can be detected. The other, the perfect binary Golay code G23 has codewords of length 23 and is obtained from the extended binary Golay code by deleting one co-ordinate position. In standard code notation the codes have parameters [24, 12, 8] and [23, 12, 7] corresponding to the length of the codewords, the dimension of the code and the minimum Hamming distance between two codewords respectively. In mathematical terms, the extended binary Golay code, G24 consists of a 12-dimensional subspace W of the space V=F224 of 24-bit words such that any two distinct elements of W differ in at least eight coordinates. By linearity, the distance statement is equivalent to any non-zero element of W having at least eight non-zero coordinates. The possible sets of non-zero coordinates as w ranges over W are called code words. In the extended binary Golay code, all code words have the Hamming weights of 0, 8, 12, 16, or 24. Up to relabeling coordinates, W is unique. The perfect binary Golay code, G23 is a perfect code. That is the spheres of radius three around code words form a partition of the vector space.

Codeword Structure: A codeword is formed by taking 12 information bits and appending 11 check bits which are derived from a modulo-2 division, as with the CRC. Golay [23, 12] Codeword. The common notation for this structure is Golay [23, 12], indicating that the code has 23 total bits, 12 information bits, and 23- 12=11 check bits. Since each codeword is 23 bits long, there are 223, or 8,388,608 possible binary values. However, since each of the 12-bit information fields has only one corresponding set of 11 check bits, there are only 212, or 4096, valid Golay code words.

Check bits	Information bits
XXX XXXX XXXX	XXXX XXXX XXXX

Golay [24,12] Codeword

Parity bit	Check bits	Information bits
X	XXX XXXX XXXX	XXXX XXXX XXXX

Figure 1: Block Diagram of Golay Code

The binary Golay code leads us to the extended Golay code. Codes can be easily extended by adding an overall parity check to the end of each code word.

This extended Golay Code can be generated by the 12 × 24 matrix G = [I_{12} | A], where I_{12} is the 12 × 12 identity matrix and A is the 12 × 12 matrix

$$A=\begin{bmatrix}0&1&1&0&1&1&1&1&1&1&1&1\\1&1&1&0&1&1&1&0&0&0&1&0\\1&1&0&1&1&1&0&0&0&1&0&1\\1&0&1&1&1&0&0&0&1&0&1&1\\1&1&1&1&0&0&0&1&0&1&1&0\\1&1&1&0&0&0&1&0&1&1&0&1\\1&1&0&0&0&1&0&1&1&0&1&1\\1&0&0&0&1&0&1&1&0&1&1&1\\1&0&0&1&0&1&1&0&1&1&1&0\\1&0&1&0&1&1&0&1&1&1&0&0\\1&1&0&1&1&0&1&1&1&0&0&0\\1&0&1&1&0&1&1&1&0&0&0&1\end{bmatrix} \qquad (2)$$

The binary linear code with generator matrix G is called the extended binary Golay code and will be denoted by G24.

The extended Golay code has a minimum distance of 8. Unlike the (23, 12) code, the extended Golay code is not perfect, but simply quasi perfect.

Properties of the extended binary Golay code

- The length of G24 is 24 and its dimension is 12.
- A parity-check matrix for G24 is the 12 × 24 matrix H = [A | I_{12}].
- The code G24 is self-dual, i.e., G⊥ 24 = G24.
- Another parity-check matrix for G24 is the 12 × 24 matrix H0 = [I_{12} | A] (= G).
- Another generator matrix for G24 is the 12 × 24 matrix G0 = [A | I_{12}] (= H).
- The weight of every codeword in G24 is a multiple of 4.
- The code G24 has no codeword of weight 4, so the minimum distance of G24 is d = 8.
- The code G24 is an exactly three-error-correcting code.

Table 1: Comparison Result of Different Types of Binary Code

Binary Code	Error Detection	Error Correction	Efficiency	Distance
Single Parity-Check Bit Code	1	1	80-85%	low
Hadamard Code	2	1	50%	Low
Hamming Code	2	1	85-90%	low
Convolution Code	2	2	85-90%	Medium
Cyclic Codes	3	2	75-80%	Medium
Golay Codes	7	3	70-75%	Large
Extended Golay Codes	8	3	80-85%	Very Large

IV. CONCLUSION AND FUTURE SCOPE

In this paper, the error detection and correction code and operation for various encoder and decoder is discussed. This encoding and decoding algorithm have been successfully applied to short block codes such as error detection and correction code. Decoding algorithm consists of syndrome measurement unit, weight measurement unit and weight constraint.

The purpose of this study is to review the published encoding and decoding models in the literature and to critique their reliability effects. We will try to reduce the area, maximum combinational path delay (MCPD) of decoding algorithm of error detection and correction code.

REFERENCES

[1] Kristjane Koleci, Paolo Santini, Marco Baldi, Franco Chiaraluce, Maurizio Martina And Guido Masera, “Efficient Hardware Implementation of the LEDAcrypt Decoder”, IEEE Access 2021.

[2] P. Santini, M. Battaglioni, M. Baldi, and F. Chiaraluce, “Analysis of the error correction capability of LDPC and MDPC codes under parallel bit-flipping decoding and application to cryptography,”, IEEE Trans. Communication, vol. 68, no. 8, pp. 4648_4660, Aug. 2020.

[3] J. Hu, M. Baldi, P. Santini, N. Zeng, S. Ling, and H. Wang, “Lightweight key encapsulation using LDPC codes on FPGAs”, IEEE Trans. Comput., vol. 69, no. 3, pp. 327_341, Mar. 2020.

[4] D. Zoni, A. Galimberti, andW. Fornaciari, ``Efficient and scalable FPGA oriented design of QC-LDPC bit-flipping decoders for post-quantum cryptography," *IEEE Access*, vol. 8, pp. 163419_163433, 2020.

[5] K.Koleci, M. Baldi, M. Martina, and G. Masera, ``Ahardware implementation for code-based post-quantum asymmetric cryptography," in *Proc. 3rd Italian Conf. Cybersecurity (ITASEC)*, vol. 2597, Ancona, Italy, Feb. 2020, pp. 141_152.

[6] D. Zoni, A. Galimberti, and W. Fornaciari, ``Flexible and scalable FPGA oriented design of multipliers for large binary polynomials," *IEEE Access*, vol. 8, pp. 75809_75821, 2020.

[7] M. Baldi, A. Barenghi, F. Chiaraluce, G. Pelosi, and P. Santini, ``LEDAcrypt: QC-LDPC code-based cryptosystems with bounded decryption failure rate," in *Code-Based Cryptography*, M. Baldi, E. Persichetti, and P. Santini, Eds. Cham, Switzerland: Springer, 2019, pp. 11_43.

[8] Shivani Tambatkar, Siddharth Narayana Menon, Sudarshan. V, M. Vinodhini and N.S.Murty, “Error Detection and Correction in Semiconductor Memories using 3D Parity Check Code with Hamming Code”, International Conference on Communication and Signal Processing, April 6-8, 2017, India.

[9] Wenhe JIN, Shaohua WU, Erpeng YANG and Jian JIAO, “LDPC Convolutional Codes Coded Cooperation Based on Puncturing”, ISBN 978-89-968650-9-4 ICACT2017 February 19 ~ 22, IEEE 2017.

[10] Pallavi Bhoyar, “Design of Encoder and Decoder for Golay code”, International Conference on Communication and Signal Processing, April 6-8, IEEE 2017, India.

[11] Manikandan J, Shruthi S, Mangala SJ and Agrawal VK, “Design and Implementation of Reconfigurable Coders for Communication Systems”, International Conference on VLSI Systems, Architectures, Technology and Applications (VLSI-SATA) IEEE 2016.

[12] Sindhuaja Muppalla and Koteswara Rao Vaddempudi, “A Novel VHDL Implementation of UART with Single Error Correction and Double Error Detection Capability”, SPACES-2015, Dept of ECE, KL University.

[13] Satyabrata Sarangi and Swapna Banerjee, “Efficient Hardware Implementation of Encoder and Decoder for Golay Code”, IEEE Transactions On Very Large Scale Integration (VLSI) Systems 2014.

[14] Nitin Patel, Naresh Patel, " VHDL Implementation of UART with BIST capability, IEEE 4th ICCCNT July 4 - 6, 2013, Tiruchengode, India.

[15] Shumit Saha, Md. Ashikur Rahman, Amit Thakur, "Design and Implementation of a BIST Embedded High Speed RS-422 Utilized UART over FPGA”, IEEE 4th ICCCNT July 4 - 6, 2013, Tiruchengode, India.