

# A Low-Complexity Diversity-Preserving Universal Bit-Flipping Enhanced Hard Decision Decoder for Arbitrary Linear Codes

Praveen Sai Bere \*, *Student Member, IEEE*, Mohammed Zafar Ali Khan †, *Senior Member, IEEE* and Lajos Hanzo ‡, *Life fellow, IEEE*

\*Department of Electrical Engineering, Indian Institute of Technology Hyderabad, Hyderabad, India, ee19resch01001@iith.ac.in

†Department of Electrical Engineering, Indian Institute of Technology Hyderabad, Hyderabad, India, zafar@iith.ac.in

‡School of Electronics and Computer Science, University of Southampton, Southampton, United Kingdom, lh@ecs.soton.ac.uk

**ABSTRACT** V2X (Vehicle-to-everything) communication relies on short messages for short-range transmissions over a fading wireless channel, yet requires high reliability and low latency.

Hard-decision decoding sacrifices the preservation of diversity order, leading to pronounced performance degradation in fading channels. By contrast, soft-decision decoding retains diversity order, albeit at the cost of increased computational complexity.

We introduce a novel enhanced hard-decision decoder termed as the Diversity Flip decoder (DFD) designed for preserving the diversity order. Moreover, it exhibits ‘universal’ applicability to all linear block codes. For a  $\mathcal{C}(n, k)$  code having a minimum distance  $d_{\min}$ , the proposed decoder incurs a worst-case complexity order of  $2^{(d_{\min}-1)} - 1$ . Notably, for codes having low  $d_{\min}$ , this complexity represents a significant reduction compared to the popular soft and hard decision decoding algorithms. Due to its capability of maintaining diversity at a low complexity, it is eminently suitable for applications such as V2X (Vehicle-to-everything), IoT (Internet of Things), mMTC (Massive Machine type Communications), URLLC (Ultra-Reliable Low Latency Communications) and WBAN (Wireless Body Area Networks) for efficient decoding with favorable performance characteristics. The simulation results provided for various known codes and decoding algorithms validate the performance versus complexity benefits of the proposed decoder.

**INDEX TERMS** Diversity methods; FEC (Forward error correction); Block Code.

## I. Introduction

THE adoption of Vehicle-to-everything (V2X) technology, which includes V2V (vehicle-to-vehicle), V2N (vehicle-to-network communications), V2I (vehicle-to-infrastructure) and V2P (vehicle-to-pedestrian) significantly enhances road safety, traffic flow efficiency, as well as the availability of entertainment and information services. ITS (Intelligent transportation systems) can be advanced through the implementation of V2X communications, facilitating cooperative operations [1]–[4].

The variable length of periodical broadcast message payloads exchanged between two User Equipment (UE) facil-

itating V2X applications typically falls within the range of 50 to 300 bytes. In the case of event-triggered messages, the message size can extend to a maximum of 1200 bytes. The stringent latency requirements of V2V/V2P applications mandate a maximum end-to-end latency of 100 milliseconds between two UE regardless communicating directly or through a Roadside Unit (RSU). Similarly, for V2I applications, the maximum acceptable latency between a UE and an RSU is 100 milliseconds [5]–[7].

The 3GPP standardization body has established three principal service categories for 5G networks [8]. One of these categories is the enhanced mobile broadband (eMBB) mode,

which caters to bandwidth-intensive applications such as high-resolution video streaming, augmented reality and virtual reality. Another category is mMTC (massive machine-type communication), which aims for accommodating a vast number of machine-type devices while ensuring optimal power efficiency for a prolonged battery recharge period. The third service category, URLLC (ultra-reliable and low latency communication), focuses on supporting time-sensitive applications and services requiring minimal delays.

Notably, mMTC and URLLC application services operate with shorter block lengths, representing distinct segments within these service categories.

The emergence of 5G NR technology has opened up new possibilities for Machine-Type Communication (MTC) services, which are anticipated to play a pivotal role in the Internet of Things (IoT) [9], especially in the context of the Industrial Internet of Things (IIoT) [10]. These MTC services can be categorized into two main types: mMTC (massive machine-type communications) [11] and URLLC (ultra-reliable low-latency communications) [12]. These categories delineate the diverse nature of MTC applications, catering for high-volume connectivity in the case of mMTC and stringent reliability as well as low latency requirements in the case of URLLC [12]–[14]. Short block lengths are a common feature in the realm of both mMTC and URLLC services.

mMTC applications, operating with short block lengths, necessitate low power consumption to prolong the battery recharge period, considering that a majority of these devices will operate on battery power. The need for an efficient decoder arises for mMTC applications relying on short block lengths. By contrast, URLLC must achieve both ultra-low latency and ultra-high reliability, which often pose conflicting demands. Nevertheless, to approach Shannon's capacity, legacy systems predominantly relied on coding schemes that encode information bits into longer codewords except for short control messages. But using longer codewords increases the latency jeopardizing the requirement of URLLC [15].

Maximum-minimum distance codes like BCH and Reed-Solomon codes developed back in the 1960s, have found renewed significance in the context of URLLC applications [16]–[20]. These codes enable the assignment of short codewords to information words, but require theoretically shortened mother codes for creating diverse code rates. On the other hand, RLCs (Random linear codes) offer flexibility in terms of their code rates but pose computational challenges [21], [22]. However, conventional hard-decision decoders or algebraic decoders of these short codes exhibit inadequate performance in practical scenarios involving fading channels due to their inability to maintain full diversity order<sup>1</sup> of  $d_{min}$

<sup>1</sup>The diversity order [23] is defined as

$$\text{Diversity order} = \lim_{SNR \rightarrow \infty} \frac{-\log[P_e(SNR)]}{\log[SNR]}$$

. Soft-decision maximum-likelihood decoding (MLD) has the potential to preserve diversity over fading channels. However, its computational complexity is increased, rendering it impractical for light-weight real-world applications. Further more, the provision of soft-information may also be limited due to the computational complexity associated with soft-output detectors. Additionally, the iterative exchange of soft information between the detector and decoder tends to increase the processing latency. Therefore, an essential demand emerges for conceiving an efficient hard-decision decoder that guarantees diversity preservation in fading channels at a low complexity and low latency.

Numerous decoding techniques have been proposed in the literature, as indicated by the references [25]–[42]. Within this context, Information Set Decoding (ISD) is a notable topic, as evident from references [43]–[49]. Bossert *et al.* [50] specifically explore the application of ISD to BCH codes for transmission over a Binary Symmetric Channel (BSC). However, it is essential to highlight that many of these decoding techniques exhibit high computational complexity and do not guarantee the preservation of diversity in fading channels. On the other hand, the soft Maximum Likelihood Decoding (MLD) of Conway and Sloane [51] maintains diversity, but suffers from a high computational complexity, which grows exponentially with the message length  $k$ .

As a further advance, Chase introduced a class of algorithms that preserve diversity [52]. However, these algorithms depend on algebraic decoders and cannot operate as stand-alone universal decoders.

The recently introduced hard-decision GRAND (Guessing Random Additive Noise Decoding) solution of Duffy and Medard [53], [54] allows for universal decoding of linear codes. The GRAND technique is particularly efficient for high-rate codes with short block lengths [53]. However, being a hard decision decoder, the GRAND approach fails to maintain diversity over fading channels. Numerous soft variants of GRAND have been introduced in the literature, as detailed in [55]–[59]. Abbas *et al.* proposed Fading-GRAND [60] for fading channels. In the same scenario, GRAND with pseudo-soft information [61] is proposed which relies on pseudo-soft information. Chatzigeorgiou *et al.* proposed Symbol level GRAND in [62], [63], while [64] recommends employing Random Linear Codes (RLCs) along with GRAND for Ultra-Reliable Low Latency Communication (URLLC) over massive Multiple Input Multiple Output (mMIMO) systems. Given the inherent high complexity of the GRAND technique [65], the Hybrid Metric-based GRAND (HMGRAND) has been introduced as a viable alternative for Cyclic Redundancy Check (CRC) codes commonly employed in IoT applications [65]. This method utilizes an algebraic decoder specifically designed for CRC

The maximum diversity order that is obtainable by soft-decision decoding for a code  $C(n, k)$  with minimum distance  $d_{min}$  in a fading channel is  $d_{min}$  [24].

codes, making it suitable for IoT applications. However, it is worth noting that these decoders do not provide any insight into diversity preservation.

Patel *et al.* [24], [66] introduced flip decoding (FD) as a powerful technique for Parity Check (PC) codes also known as Single Parity Check codes (SPC) [67]. FD has been shown to attain a second-order diversity for parity check codes, akin to that derived from soft decision decoding. Notably, FD exhibits a decoding complexity that scales linearly with the block length  $n$ . This breakthrough allows PC codes, traditionally used for error detection, to be effectively harnessed for error correction in Rayleigh fading channels. The potential of FD in achieving dual-radio diversity using PC codes was explored by Ahmed in [68], wherein PC codes are applied above the MAC layer and decoding is also carried out above the MAC layer to glean the benefits of both offloading and duplication. The employment of PC codes at the MAC layer presents a solution to the challenge of combining offloading and duplication for multi-radios [69]–[82] for dual radios. Note that soft decoding is cumbersome above the MAC layer [83] and hence extending the coding and decoding above the MAC layer for more than two radios has certain requirements that are listed below:

- 1) Low complexity encoding: This requirement has restricted the use of codes to cyclic codes (including CRC codes) that have efficient low-complexity implementation using shift registers.
- 2) Low complexity decoding limits the decoding to hard decision techniques, since passing the soft values to the MAC or to the layers above the MAC is practically impossible.
- 3) Availability of RSSI: Since, all standards pass RSSI values to the MAC layer, this could be used for enhancing the hard decision decoding.

Independently, the use of flip decoding was further expanded to Hamming codes in [84] and CRC codes in [85]. These extensions markedly enhance the error-correction capability of Hamming and CRC codes in contrast to conventional hard-decision decoding approaches, all the while preserving the diversity order. Notably, the decoding complexity for both code types retains linearity in  $n$ .

Motivated by these benefits, we extend flip decoding to a broad family of linear codes, rendering it a ‘universal’ decoder that can be employed for any linear block code. The proposed decoder, which we have renamed as diversity flip decoder (DFD) to highlight its diversity preserving nature, attains an identical diversity order ( $d_{\min}$ ) to soft-decision decoding despite its low complexity. This makes the proposed diversity flip decoder especially well-suited for decoding in fading channels, where diversity plays a pivotal role in alleviating the impact of fading.

### A. Our Main Contributions

Our novel contributions are boldly contrasted to the literature in Table I and further detailed as follows:

- **Universal Applicability:** We introduce a universal DFD technique designed to preserve diversity in fading channels for all linear block codes.
- **Enhanced Error Correction:** We demonstrate that the proposed DFD significantly improves the error correction capabilities of linear block codes. Specifically, it can correct up to  $(d_{\min} - 1)$  errors, but not all  $(d_{\min} - 1)$  errors.
- **Preserved Diversity Order:** We show that our proposed decoder achieves a diversity order equal to ‘ $d_{\min}$ ’, aligning it with the diversity order of soft-decision decoding techniques, which is a critical aspect for improving reliability in fading channels.
- **Low Complexity:** The inherent advantage of the proposed DFD technique is its low computational complexity, when compared to well-established soft-decision and hard-decision decoders. This paves the way for its practical implementation.
- **Theoretical and Empirical Validation:** We provide comprehensive theoretical analyses complemented by our simulations to confirm its diversity order.
- **Enabling coding/decoding at MAC and above layers:** The proposed enhanced hard-decision decoder is particularly suitable for Multi-radios to achieve the benefits of both offloading and duplication.
- **Application Versatility:** The above benefits position the proposed DFD technique as an ideal solution for short-block length applications in V2X, WBAN, IoT, mMTC, and URLLC scenarios, where low complexity and high performance are paramount.

In Table I, the Chase decoder, OSD, ISD, ORBGRAND offer a correction capability higher than  $t$  ( $= \lfloor \frac{d_{\min}-1}{2} \rfloor$ ) since they utilize soft information. By contrast, GRAND is capable of correcting up to a maximum of  $t$  errors. The OSD, ISD, GRAND, and ORBGRAND decoders are considered universal, as they are applicable to all types of codes. However, Chase decoders lack universality, since they are unsuitable for codes like Parity check codes or CRC codes due to their reliance on specific hard-decision decoders. Consequently, Chase decoders cannot function as standalone decoders, unlike the OSD, ISD, GRAND, and ORBGRAND decoders. Chase decoders maintain diversity over fading channels, whereas GRAND does not preserve the diversity order under fading conditions. For OSD, ISD, and ORBGRAND, the preservation of diversity order is not guaranteed at all times.

### B. Structure of the Paper

In Section II, we commence with an overview of the system setup, followed by the relevant concepts. The state-of-the-art is briefly summarized in Section C. Section III introduces

TABLE I: Distinguishing Our New Contributions from the State-of-the-Art

Attribute	Decoder type					Proposed DFD	Proposed EDFD
	Chase [52]	OSD [35], [46]	ISD [18], [43], [44]	GRAND [53]	ORBGRAND [58], [60], [61]		
Enhanced Error correction ( $> t$ )	✓	✓	✓		✓	✓	✓
Universal decoder		✓	✓	✓	✓	✓	✓
Stand-alone decoder		✓	✓	✓	✓	✓	✓
Preservation of diversity	✓					✓	✓
Low Complexity						✓	✓
Suitability for decoding at MAC and above MAC layers				✓		✓	✓

our diversity-preserving DFD technique, while Section IV presents the derivation of an upper bound on the probability of error and the diversity order of DFD. Section V provides our simulation results. In Subsection A, DFD's achievement of the diversity order  $d_{min}$  is demonstrated. In Subsection B, the DFD's performance for different codeword lengths and code rates are presented, while Subsection C offers our performance versus complexity analysis. Section VI introduces the Extended Diversity Flip Decoder (EDFD), which represents a further enhancement in performance. Subsequently, the simulation results for the EDFD are detailed in the subsequent section, Section VII. Finally, our conclusions are drawn in Section VIII.

## II. Preliminaries

### A. System setup

Consider a binary linear block code  $\mathcal{C}(n, k)$  having a parity check matrix (PCM)  $\mathbf{H}_{(n-k) \times n}$  and a generator matrix (GM)  $\mathbf{G}_{k \times n}$ . Here,  $n$  denotes the codeword length and  $k$  represents the dimension of the code  $\mathcal{C}(n, k)$ , which possesses a minimum distance of  $d_{min}$ .

We adopt the following notation: a matrix is denoted by  $\mathbf{X}$ , while a vector is denoted by  $\mathbf{y}$ ; a row vector with a length of  $n$  is denoted by  $\mathbf{y}^n$ ;  $\mathbf{0}^n$  indicates a row vector of zeros with a length of  $n$ , and  $y$  stands for a scalar;  $w_H(\mathbf{y})$  denotes the Hamming weight of a binary vector  $\mathbf{y}$ ;  $\mathbf{q} \oplus \mathbf{p}$  formulates the binary XOR operation of the corresponding elements of the pair of binary vectors  $\mathbf{q}, \mathbf{p}$ , where we have:

$$\mathbf{q} \oplus \mathbf{p} = [q_1 \oplus p_1 \quad q_2 \oplus p_2 \quad q_3 \oplus p_3 \quad \cdots \quad q_n \oplus p_n]_{1 \times n}$$

with  $q_j, p_j \in \{0, 1\}$ .

Let  $\mathbf{c} = [c_1 \quad c_2 \quad c_3 \quad \cdots \quad c_n]_{1 \times n} \in \mathcal{C}(n, k)$  be a legitimate transmitted codeword. In the case of a practical scenario involving a Rayleigh fading channel, BPSK modulation is assumed for simplicity. Under BPSK modulation,  $\mathbf{c}$  is mapped to a symbol sequence  $\mathbf{s} = [s_1 \quad s_2 \quad s_3 \quad \cdots \quad s_n]_{1 \times n}$ , where we have  $s_j = (-1)^{c_j}$ . Thus,  $s_j \in \{-1, 1\}, \forall j$ . Then, the received signal can be represented as:

$$y_j = h_j s_j + w_j, \quad 1 \leq j \leq n, \quad h_j \geq 0, \quad (1)$$

where  $w_j$  represents statistically independent complex Gaussian random variables with zero mean and variance of  $N_0$ . The coefficients  $h_j$  denote identically distributed, independent Rayleigh fading coefficients, indicating a fast-fading channel assumption. In the context of a slow-fading channel, the codewords are assumed to be interleaved ideally. Let  $\mathbf{h} = [h_1 \quad h_2 \quad \cdots \quad h_n]_{1 \times n}$  denote the Channel State Information (CSI) vector. Additionally, it is assumed that the receiver has perfect knowledge of the CSI.

The received vector is denoted by  $\mathbf{y} = [y_1 \quad y_2 \quad \cdots \quad y_n]_{1 \times n}$ . Applying a hard decision to  $\mathbf{y}$ , the received vector is represented as  $\mathbf{r} = [r_1 \quad r_2 \quad \cdots \quad r_n]_{1 \times n}$ .

### B. Known Results of Diversity

The linear block code  $\mathcal{C}(n, k)$  with minimum distance  $d_{min}$  and error correcting capability  $t$  exhibits a diversity order of  $(t + 1) = \lfloor \frac{d_{min}-1}{2} \rfloor + 1$  under hard decision decoding, while soft decision decoding achieves a diversity order of  $d_{min}$  [24].

While hard decision decoding of  $\mathcal{C}(n, k)$  offers lower complexity than soft decision decoding, it sacrifices diversity order, leading to significant performance degradation in fading channels. On the other hand, soft decision maximum likelihood decoding delivers excellent performance, but involves a comparison of  $2^k$  legitimate codewords, resulting in high complexity. As the value of  $k$  increases, the complexity of soft decision ML decoding grows exponentially.

### C. Previous Relevant Work

In this subsection, we present a comprehensive overview of Chase, OSD, ISD (Information Set Decoding), along with recent research related to GRAND (Guessing Random Additive Noise Decoding) decoders. We classify them into "guessing" noise or codeword type.

#### Definition 1:

Guessing codeword decoders are the decoders that search over codewords.

TABLE II: Categorizing the decoders

Decoder	Type
Chase	Guessing codeword type
OSD	Guessing codeword type
ISD	Guessing codeword type
GRAND	Guessing noise type
ORBGAND	Guessing noise type
Proposed DFD	Guessing noise type
Proposed EDFD	Guessing noise type

**Definition 2:**

Guessing noise decoders are the decoders that search over different noise patterns.

In Table II, various decoders are categorized into two groups: ‘Guessing noise type’ and ‘Guessing codeword type’.

1) Flip Decoding for Parity check codes

The idea of Flip decoding for PC codes appeared first in [66]. This methodology entails flipping the bit linked with the minimum Channel State Information (CSI) value whenever a parity mismatch arises at the receiver. Through this Flip Decoder (FD) technique, PC codes have showcased a notable diversity order of 2 in fading channels.

2) Chase decoders

In 1972, Chase introduced a class of decoding algorithms that leverage channel measurement information [52]. Here’s a concise overview of the Chase decoding algorithms: Start with the received sequence denoted as  $\underline{y}$ . Introduce a test pattern, denoted as  $\underline{T}$ , added to  $\underline{y}$  to produce a new sequence,  $\underline{y}'$ . For each  $\underline{y}'$ , utilize a hard-decision binary decoder to derive a new sequence,  $\underline{y}''$ . Let  $\underline{z}'$  represent the corresponding error pattern. Then, the effective error pattern is determined by

$$\underline{z}_T = \underline{T} \oplus \underline{z}'.$$

Ultimately, the effective error pattern ( $\underline{z}_T$ ) having the lowest analog weight [52] is chosen as the definitive error pattern, and its associated codeword is designated as the decoded codeword.

Chase presented three algorithms characterized by a progressively diminishing count of error patterns ( $\underline{T}$ ) within the specified error pattern set.

*Algorithm-1:* This algorithm encompasses all error patterns having a binary Hamming weight of  $\lfloor \frac{d_{min}}{2} \rfloor$ . Consequently, a total of  $\binom{n}{\lfloor \frac{d_{min}}{2} \rfloor}$  error patterns are included in this approach, which can be substantial in size.

*Algorithm-2:* This approach involves error patterns ( $\underline{T}$ ) where all combinations of 1’s are positioned within the

$\lfloor \frac{d_{min}}{2} \rfloor$  lowest channel measurement information values. Consequently,  $2^{\lfloor \frac{d_{min}}{2} \rfloor}$  test error patterns are taken into account in this method.

*Algorithm-3:* In this approach, the complexity is further reduced. It involves considering test patterns with a Hamming weight of  $i$ , where  $i$  represents the number of 1’s positioned in slots corresponding to the  $i$  lowest channel measurement values. The values of  $i$  encompass:

$$\begin{aligned} i &= 0, 1, 3, \dots, d_{min} - 1, & \text{when } d_{min} \text{ is even.} \\ i &= 0, 2, 4, \dots, d_{min} - 1, & \text{when } d_{min} \text{ is odd.} \end{aligned}$$

Consequently, the total count of error patterns considered will be  $\lfloor (\frac{d_{min}}{2}) + 1 \rfloor$ , resulting in a significant reduction in the number of error patterns.

**Remark 1:**

It’s noteworthy that Chase decoders aren’t universally applicable since their decoding relies on the specific hard-decision decoding algorithm of the code.

3) OSD

Let  $\underline{\alpha} = [\alpha_1 \ \alpha_2 \ \dots \ \alpha_n]$  be a reliable vector holding the reliabilities of all  $n$  bits of the received codeword  $\underline{r}$ . In OSD, the reliabilities are ordered in descending order to get  $\underline{\alpha}_{sort} = [\alpha_{(1)} \ \alpha_{(2)} \ \dots \ \alpha_{(n)}]$ . The corresponding permutation function is  $\lambda_1$ . Then,  $\lambda_1$  is applied to  $\underline{r}$  to get  $\underline{r}'$ . The same  $\lambda_1$  is applied to the columns of the generator matrix  $G$  to get  $G'$ . Then, the first  $k$  independent columns of  $G'$  are identified and selected as the first  $k$  columns of  $G''$ . The remaining columns of  $G'$  will be the next  $(n - k)$  columns of  $G''$ . Let this permutation function be  $\lambda_2$ . The matrix  $G''$  is converted into a systematic form to get  $G_{OSD}$ . Then,  $\lambda_2$  is applied to  $\underline{r}'$  in order to get  $\underline{r}''$ . The first  $k$  bits in  $\underline{r}''$  are denoted by  $\underline{r}''_M$  and the remaining bits are denoted by  $\underline{r}''_L$ . In OSD, Test Error Patterns (TEPs) of length  $k$  bits are employed for reprocessing. For order- $i$  reprocessing, TEPs ( $\underline{e}$ ) up to Hamming weight of  $i$  are considered. The codeword is reprocessed as follows

$$\tilde{\underline{c}}_e = (\underline{r}''_M \oplus \underline{e}) \cdot G_{OSD}.$$

Let  $\tilde{\underline{\alpha}} = [\tilde{\alpha}_1 \ \tilde{\alpha}_2 \ \dots \ \tilde{\alpha}_n] = \lambda_2[\lambda_1(\underline{\alpha})]$ . The weighted Hamming distance is defined as follows:

$$d^{(W)}(\underline{r}'', \tilde{\underline{c}}_e) \triangleq \sum_{\substack{1 \leq i \leq n \\ r''_i \neq \tilde{c}_{e_i}}} \tilde{\alpha}_i.$$

In OSD, a particular codeword corresponding to the minimum weighted Hamming distance is chosen as an optimal codeword ( $\tilde{\underline{c}}_{opt}$ ). Then, the appropriate inverse permutations are applied to  $\tilde{\underline{c}}_{opt}$  to get the final decoded codeword ( $\hat{\underline{c}}_{opt}$ )

$$\hat{\underline{c}}_{opt} = \lambda_1^{-1}[\lambda_2^{-1}(\tilde{\underline{c}}_{opt})].$$



#### 4) ISD

ISD [18], [43] uses specific bit positions  $(i_0, i_1, i_2, \dots, i_{k-1})$  to calculate a systematic basis<sup>2</sup> [18] ( $B = 0, 1, 2, \dots, k-1$ ) and if these positions contain linearly dependent columns, it extends the search until a systematic basis is found. Using this basis, ISD then encodes data with received bits  $r_0, r_1, r_2, \dots, r_{k-1}$  as information. It is preventing decoding failure. ISD can also become a list decoder, flipping information bits and choosing the codeword having the smallest Hamming distance.

#### 5) GRAND and GRANDAB

The GRAND [53] technique capitalizes on the observation that as the channel conditions improve, the set of potential error patterns capable of altering a legitimate transmitted codeword becomes smaller. This insight enables a shift in focus from searching for the most likely transmitted codeword within the codebook to estimating the most probable error pattern.

To determine whether an error pattern corresponds to a valid codeword, a code-book membership test is employed. GRAND organizes the prospective error patterns in descending order of likelihood and sequentially applies the membership test to each error pattern. The algorithm terminates and returns the first error pattern that successfully passes the membership test.

In contrast to GRAND, GRANDAB [59] not only halts at the first occurrence of an affirmative response but also terminates and indicates an erasure when the number of codebook queries exceeds a certain threshold known as Abandonment bit (ABB).

#### 6) ORBGRAND

Consider the transmission of  $\underline{c}$  over a binary-input AWGN channel, with  $\underline{r}$  as the received vector. The codeword bits can be sorted in ascending order of reliability based on  $\underline{r}$ , and a vector  $\underline{s}$  is used to record the reliability order of each bit. For an error pattern vector  $\underline{z}$ , the logistic weight  $w_l$  is defined as  $w_l = \sum_{i=0}^{n-1} (s_i \times z_i)$ . It follows that  $w_l(z^{k_1}) < w_l(z^{k_2})$  for  $k_1 < k_2$ . Smaller values of  $w_l(z^k)$  indicate higher likelihoods of the corresponding error patterns.

#### 7) Fading-GRAND

The fading-GRAND of [60] leverages channel state information to identify a set of bit indices denoted as  $\mathcal{I}$ . This set is constructed based on a given threshold value ( $\Delta$ ). The

<sup>2</sup>systematic basis is the linearly independent  $k$  columns. For example a systematic code  $\mathcal{C}(n, k)$  with generator matrix  $G_{k \times n}$  is considered. Let the set  $P \subset \{0, 1, 2, \dots, n-1\}$  represent a set of  $k$  distinct positions. If the submatrix  $G_P$  comprising columns  $j \in P$ , achieves full rank  $k$ ,  $P$  is termed an information set. These positions uniquely determine the remaining  $n-k$  positions in any codeword.

expression for  $\Delta$  is defined as  $\Delta = m \left( \frac{E_b}{N_0} \right) + b$ , where  $m$  and  $b$  represent the optimal values that yield superior performance for different codes, as specified in [60]. In the next step, Test Error Patterns (TEPs) are generated by ensuring that the positions of 1s in the TEPs exclude the positions specified in the set  $\mathcal{I}$ .

#### 8) GRAND using pseudo-soft information

The method described in [61] introduces an approach where the received bits are arranged according to their pseudo-soft information, which incorporates both the channel state information (CSI) and the colored additive noise (CAN) statistics, rather than conventional soft information. Subsequently, the remaining steps of the GRAND technique are applied without any modifications.

### III. Proposed Universal Diversity Flip Decoder

This section presents the proposed Diversity Flip decoder (DFD) conceived for all linear block codes. The essence of the proposed DFD lies in its operational principle. Within fading channels, the fading envelope has a more pronounced influence on errors than the Additive White Gaussian Noise (AWGN). Consequently, the DFD leverages the knowledge of Channel State Information (CSI) to rectify the errors by selectively flipping the bits associated with the lowest CSI values. These bits are the ones most likely to be erroneous, enabling their targeted flipping within the corrupted codeword in order to restore the integrity of the transmitted codeword.

#### A. The Diversity Flip Decoder

Consider the linear block code denoted as  $\mathcal{C}(n, k)$ , possessing a minimum distance of  $d_{\min}$ . Let  $\underline{c} = [c_1 \ c_2 \ \dots \ c_n]_{1 \times n} \in \mathcal{C}$  represent the transmitted codeword corresponding to the information word  $\underline{m} = [m_1 \ m_2 \ \dots \ m_k]_{1 \times k}$ . After a hard-decision process at the receiver, the received word is denoted as  $\underline{r} = [r_1 \ r_2 \ \dots \ r_n]_{1 \times n}$ . Furthermore, the vector  $\underline{h} = [h_1 \ h_2 \ \dots \ h_n]_{1 \times n}$  is defined as the collection of absolute values characterizing the channel state information (CSI).

In the flip decoding process, we sort the elements of  $\underline{h}$  in ascending order (similar to OSD [86]), resulting in the sorted vector denoted as  $\underline{h}_{\text{sort}} = [h_{(1)} \ h_{(2)} \ \dots \ h_{(n)}]_{1 \times n}$ . Let ' $\pi$ ' represent the corresponding permutation function. Applying the same permutation function ' $\pi$ ' to the vector  $\underline{r}$  yields the vector  $\underline{r}_{\text{sort}} = [r_{(1)} \ r_{(2)} \ \dots \ r_{(n)}]_{1 \times n}$ . Thus, we can express it as:

$$\begin{aligned} \underline{h}_{\text{sort}} &= \pi(\underline{h}), \\ \underline{r}_{\text{sort}} &= \pi(\underline{r}), \\ \Rightarrow \underline{r} &= \pi^{-1}(\underline{r}_{\text{sort}}). \end{aligned}$$

Define

$$d = d_{\min} - 1. \quad (2)$$

Then the initial  $d$  bits in  $\mathbf{r}_{\text{sort}}$  represent the least reliable bits (LRB), while the remaining bits are referred to as the most reliable bits (MRB). This can be expressed as follows:

$$\mathbf{r}_{\text{lrB}} = [r_{(1)} \ r_{(2)} \ r_{(3)} \ \cdots \ r_{(d)}]_{1 \times d},$$

$$\mathbf{r}_{\text{mrB}} = [r_{(d+1)} \ r_{(d+2)} \ r_{(d+3)} \ \cdots \ r_{(n)}]_{1 \times (n-d)}.$$

Consequently, we have:

$$\mathbf{r}_{\text{sort}} = [\mathbf{r}_{\text{lrB}} \ \mathbf{r}_{\text{mrB}}].$$

Subsequently, we embark on an exploration of all conceivable patterns of flipping the least reliable bits (LRB). Each flip vector ( $\mathbf{f}_{\text{pv}}^n$ ) is represented as a row vector of the same size as the codeword  $\mathbf{c}^n$ . We formally define the set of flip vectors as  $\Phi$ , denoted as:

$$\Phi = \left\{ \mathbf{f}_{\text{pv}}^n \mid \mathbf{f}_{\text{pv}}^n = [\mathbf{f}_{\text{p}}^{d(i)} \ \mathbf{0}^{n-d}] \right\} \quad (1 \leq i \leq 2^d - 1).$$

For a given decimal number  $i$ , the flip pattern vector, denoted as  $\mathbf{f}_{\text{p}}^{d(i)}$ , is constructed such that the least significant bit (LSB) takes the first position when converting the decimal number  $i$  to a binary vector of length  $d$ . Additionally,  $\mathbf{f}_{\text{pv}}^n$  is referred to as the ‘flip vector’. To illustrate this, consider the BCH(15,7,5) code where  $i = 3$ ,  $d = 4$ , and  $n = 15$ . We have:

$$\mathbf{f}_{\text{p}}^{d(3)} = [1 \ 1 \ 0 \ 0]$$

$$\mathbf{f}_{\text{pv}}^n = [\mathbf{f}_{\text{p}}^{d(3)} \ \mathbf{0}^{11}]$$

$$= [1 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0]$$

Please be aware that for a given ‘ $d = d_{\min} - 1$ ’, the set of flip patterns remains fixed and does not change with the type of code, as long as the minimum distance ‘ $d_{\min}$ ’ is constant. The composition of the final set of flip vectors is determined by appending corresponding zero vectors based on the codeword length ‘ $n$ ’. Therefore, the creation of the flip vector set is straightforward, has low complexity, and does not necessitate the repeated generation of flip vectors for each code type employed.

Let  $\Phi(l)$  denote the  $l^{\text{th}}$  element of the set  $\Phi$ . The cardinality of  $\Phi$  is  $2^d - 1$ . The Flip decoding technique operates as follows: Each flip vector is sequentially selected from the set  $\Phi$ , and it is added (XORed) to the sorted received vector,  $\mathbf{r}_{\text{sort}}$ , resulting in  $\mathbf{r}_{\text{t}}$ . Then, the inverse permutation function of  $\mathbf{r}_{\text{t}}$  is applied to obtain the resultant vector  $\mathbf{r}_{\text{o}}$ . This can be expressed as:

$$\mathbf{r}_{\text{t}} = \mathbf{r}_{\text{sort}} \oplus \mathbf{f}_{\text{pv}},$$

$$\mathbf{r}_{\text{o}} = \pi^{-1}(\mathbf{r}_{\text{t}}).$$

Subsequently, the syndrome vector ( $\mathbf{S}_{\text{y}}$ ) is computed for  $\mathbf{r}_{\text{o}}$  using the matrix  $\mathbf{H}$  as  $\mathbf{S}_{\text{y}} = \mathbf{r}_{\text{o}} \mathbf{H}^T$ . If the resultant syndrome ( $\mathbf{S}_{\text{y}}$ ) is non-zero, the next flip vector is selected from the set  $\Phi$  and added to the vector  $\mathbf{r}_{\text{sort}}$ . This iterative process continues until a zero syndrome is obtained. Upon obtaining

a zero syndrome, the resultant vector ( $\mathbf{r}_{\text{o}}$ ) represents the decoded codeword ( $\mathbf{c}_{\text{o}}^{\text{opt}}$ ), thereby concluding the process. However, if none of the flip vectors  $\mathbf{f}_{\text{pv}}^n$  in the set  $\Phi$  yields a valid codeword, Algorithm-1 returns the received word ( $\mathbf{r}$ ) as the decoded codeword  $\mathbf{c}_{\text{o}}^{\text{opt}}$ . Subsequently, the further decoded information word  $\mathbf{m}_{\text{o}}^{\text{opt}}$  is derived by employing the codebook  $\mathcal{C}$ .

Algorithm-1 provides the pseudo-code of the FD applicable to all linear block codes. In this algorithm, the ‘ $\sim$ ’ operator represents the logical NOT operation. The function ‘ $\text{sum}(\mathbf{S}_{\text{y}})$ ’ calculates and returns the sum of the elements in vector  $\mathbf{S}_{\text{y}}$ . Additionally, the function ‘ $\text{sort}(\mathbf{h})$ ’ yields the vector  $\mathbf{h}_{\text{sort}}$ , where the elements of  $\mathbf{h}$  are arranged in ascending order. It also provides a vector,  $\mathbf{Ix}$  which serves as the permutation function containing the permuted indices corresponding to the elements of  $\mathbf{h}$ . The functionality of ‘ $\text{sort}(\mathbf{h})$ ’ can be better understood through the following example.

$$\text{If } \mathbf{h} = [0.82 \ 1.3 \ 1.08 \ 0.09 \ 0.43 \ 1.8 \ 0.32];$$

$$[\mathbf{h}_{\text{sort}}, \mathbf{Ix}] = \text{sort}(\mathbf{h}); \text{ then,}$$

$$\mathbf{h}_{\text{sort}} = [0.09 \ 0.32 \ 0.43 \ 0.82 \ 1.08 \ 1.3 \ 1.8]; \text{ and}$$

$$\mathbf{Ix} = [4 \ 7 \ 5 \ 1 \ 3 \ 2 \ 6];$$

$$\mathbf{Ix}(1) = 4; \ \mathbf{Ix}(3) = 5.$$

#### Example 1:

The Flip decoding process is elucidated through the examination of the following illustrative example involving a BCH(15,7,5) code. Let the transmitted codeword be

$$\mathbf{c} = [1 \ 0 \ 0 \ 1 \ 1 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0 \ 0 \ 0 \ 1 \ 0]_{1 \times n}.$$

Consider the received codeword after hard-decision, along with the associated Channel State Information (CSI) vector at the receiver, denoted by

$$\mathbf{r} = [1 \ 0 \ 0 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 0 \ 0 \ 0 \ 0 \ 1 \ 0]_{1 \times 15},$$

$$\mathbf{h} = [1.0869 \ 0.7561 \ 2.496 \ 1.8351 \ 0.416$$

$$0.1256 \ 0.9395 \ 1.6002 \ 0.4133 \ 1.6239$$

$$0.0854 \ 1.1069 \ 0.817 \ 0.9698 \ 1.5772]_{1 \times 15}.$$

When the elements of  $\mathbf{h}$  are arranged in ascending order, resulting in  $\mathbf{h}_{\text{sort}}$ , along with the corresponding permutation function  $\pi$  represented as a vector  $\mathbf{Ix}$ , applying this permutation function  $\pi$  to  $\mathbf{r}$  yields  $\mathbf{r}_{\text{sort}}$ .

$$[\mathbf{h}_{\text{sort}}, \mathbf{Ix}] = \text{sort}(\mathbf{h}); \text{ then,}$$

$$\mathbf{h}_{\text{sort}} = [0.0854 \ 0.1256 \ 0.4133 \ 0.416 \ 0.7561$$

$$0.817 \ 0.9395 \ 0.9698 \ 1.0869 \ 1.1069$$

$$1.5772 \ 1.6002 \ 1.6239 \ 1.8351 \ 2.496],$$

$$\pi : \ \mathbf{Ix} = [11 \ 6 \ 9 \ 5 \ 2 \ 13 \ 7 \ 14$$

$$1 \ 12 \ 15 \ 8 \ 10 \ 4 \ 3],$$

$$\mathbf{r}_{\text{sort}} = [0 \ 1 \ 1 \ 1 \ 0 \ 0 \ 1 \ 1 \ 1 \ 0$$

$$0 \ 1 \ 0 \ 1 \ 0].$$

For a BCH(15,7,5) code, the minimum distance is  $d_{\min} = 5$ . Consequently,  $d = d_{\min} - 1 = 4$ . Hence the set of flip vectors, denoted as  $\Phi$ , is thus expressed as follows:

$$\Phi = \{ \begin{aligned} &[1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0], \\ &[0 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0], \\ &[1 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0], \\ &[0 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0], \\ &[1 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0], \\ &[0 \ 1 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0], \\ &[1 \ 1 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0], \\ &[0 \ 0 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0], \\ &[1 \ 0 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0], \\ &[0 \ 1 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0], \\ &[1 \ 1 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0], \\ &[0 \ 0 \ 1 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0], \\ &[1 \ 0 \ 1 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0], \\ &[0 \ 1 \ 1 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0], \\ &[1 \ 1 \ 1 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0], \end{aligned} \}$$

Let us now compute the syndrome for the received codeword,  $\mathbf{r}$ , resulting in  $\mathbf{S}_y = [0 \ 1 \ 1 \ 1 \ 0 \ 0 \ 1 \ 1]$ . As the syndrome is non-zero ( $\mathbf{S}_y \neq \mathbf{0}$ ), we select the first flip vector from the set  $\Phi$  and add it to  $\mathbf{r}_{\text{sort}}$ . Subsequently, we calculate the syndrome for the resultant vector as part of the valid codeword check. Thus, we have:

$$\begin{aligned} \mathbf{f}_{\text{pv}} &= \Phi(1); \\ \mathbf{r}_t &= \mathbf{r}_{\text{sort}} \oplus \mathbf{f}_{\text{pv}}; \\ &= [0 \ 1 \ 1 \ 1 \ 0 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 0] \oplus \\ &\quad [1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0] \\ &= [1 \ 1 \ 1 \ 1 \ 0 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 0]; \\ \mathbf{r}_o &= \pi^{-1}(\mathbf{r}_t); \\ &= [1 \ 0 \ 0 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 0 \ 1 \ 0 \ 0 \ 1 \ 0]; \\ \mathbf{S}_y &= \mathbf{r}_o \mathbf{H}^T; \\ &= [0 \ 1 \ 1 \ 0 \ 0 \ 0 \ 1 \ 1]. \end{aligned}$$

Since the syndrome is non-zero, we proceed by selecting the next flip vector from the set  $\Phi$  and repeat the same steps:

$$\begin{aligned} \mathbf{f}_{\text{pv}} &= \Phi(2); \\ \mathbf{r}_t &= \mathbf{r}_{\text{sort}} \oplus \mathbf{f}_{\text{pv}}; \\ &= [0 \ 1 \ 1 \ 1 \ 0 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 0] \oplus \\ &\quad [0 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0] \\ &= [0 \ 0 \ 1 \ 1 \ 0 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 0]; \\ \mathbf{r}_o &= \pi^{-1}(\mathbf{r}_t); \\ &= [1 \ 0 \ 0 \ 1 \ 1 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0 \ 0 \ 0 \ 1 \ 0]; \\ \mathbf{S}_y &= \mathbf{r}_o \mathbf{H}^T; \\ &= [0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0]. \end{aligned}$$

With the second iteration yielding a zero syndrome, indicating the validity of  $\mathbf{r}_o$  as a codeword, the process concludes. The decoded codeword ( $\mathbf{c}^{\text{opt}}$ ) is determined as  $\mathbf{r}_o$ , and the associated message bits are extracted by selecting the first  $k$  bits from  $\mathbf{c}^{\text{opt}}$ , since we considered the systematic BCH(15,7,5) code.

By systematically exploring and manipulating the least reliable bits, the flip decoder enhances the decoding performance, while maintaining diversity.

The proposed DFD achieves diversity, thereby offering excellent performance over fading channels at a very low complexity. By contrast, the original hard-decision GRAND [53] does not preserve diversity order due to its hard-decision nature, resulting in lower performance compared to DFD. Other soft-decision GRAND variants, such as ORBGRAND [58], exhibit superior performance to DFD at a significantly higher complexity. Additionally, all GRAND variants introduce complexity in two aspects: the complexity of generating the next tentatively assumed noise sequence and the complexity involved in verifying the validity of the resultant codeword through syndrome calculations.

In the case of DFD, the set of flip patterns remains fixed for a given minimum distance ( $d_{\min}$ ), regardless of the code. Flip vectors can be generated from this set by appending zeros based on the codeword length ( $n$ ). Consequently, there is no complexity associated with generating a new flip vector each time that needs to be added to the original received word. Thus, the complexity of DFD is significantly lower compared to GRAND and its variants.

To distinguish between the proposed DFD and ORBGRAND methodologies, the error patterns corresponding to DFD and ORBGRAND in case of (15,7,5) BCH code are illustrated in Figures 1a and 1b, respectively. Figure 1a displays the flip vector pattern associated with DFD, while Figure 1b showcases the error pattern of ORBGRAND up to the 15<sup>th</sup> query order. In these representations, the black color signifies the flip operation, whereas white indicates the absence of any flip.

#### IV. The Flip Decoder's Error Probability and Diversity Analysis

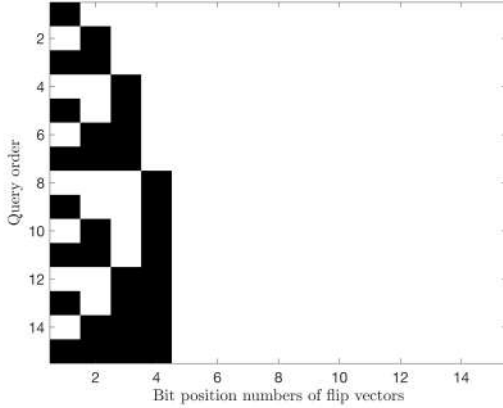
We commence by establishing an upper bound for the probability of error denoted as  $P_{FD}(\mathcal{C})$ . Subsequently, we proceed to deduce the diversity order of the DFD.

Consider the received word  $\mathbf{r}$  having a total of  $n_e$  errors. Additionally, we define  $\rho_i = h_i^2 \bar{\rho}_c$  for  $1 \leq i \leq n$ , representing the instantaneous Signal-to-Noise Ratio (SNR) per coded bit, where  $\bar{\rho}_c$  is the average SNR per coded bit.

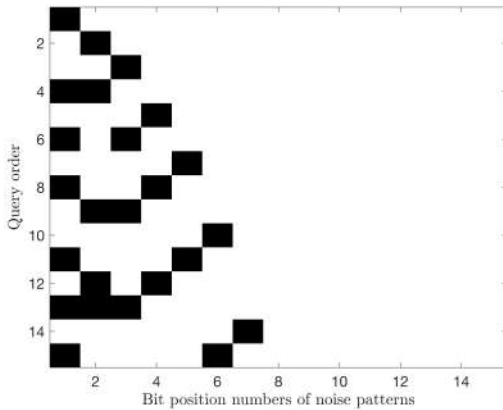
In the ensuing scenarios, the proposed decoder proves to be unsuccessful in rectifying errors within the received word.

- **Case-1:** In this scenario, where there are  $n_e = v$  errors in the received word, with  $1 \leq v \leq d$ , it is certain that at least one of these errors resides in the Most Reliable Bit





(a) Proposed DFD Flip vector pattern sequence.



(b) ORBGRAND error pattern sequence

FIGURE 1: Error patterns of DFD and ORBGRAND: BCH (15, 7).

(MRB). Consequently, the decoder becomes incapable of rectifying all errors within this context. We denote the probability of this situation by  $P_{e_v}$ .

- **Case-2:** In this particular situation, there are  $d + 1$  or more erroneous bits within the received vector  $\mathbf{r}$ . To represent this case, we use  $P_{e_{d+1}}$  to denote its probability.

The overall error probability for the proposed DFD is obtained by summing up these individual probabilities. Therefore, we have

$$P_{FD}(\mathcal{C}) = \left( \sum_{v=1}^d P_{e_v} \right) + P_{e_{d+1}}. \quad (3)$$

We now proceed with the derivation of the probability of error pertaining to Case-1, wherein the occurrence of  $v$  ( $1 \leq v \leq d$ ) errors is considered.

#### Algorithm 1: Universal Diversity Flip decoder

##### Input:

- $\Phi$ : List of Flip vectors.
- $\mathbf{r}$ : received codeword.
- $\mathbf{h}$ : CSI vector.
- $\mathbf{H}$
- $d_{\min}$
- $n_{\Phi}$ : The cardinality of the set  $\Phi$ .

##### Output:

- $\mathbf{c}_o^{\text{opt}}$ : The decoded codeword.

```

1  $\mathbf{S}_y = \mathbf{r}\mathbf{H}^T$ ;
2 if ( $\text{sum}(\mathbf{S}_y) > 0$ ) then
3    $d = d_{\min} - 1$ ;
4    $[\mathbf{h}_{\text{sort}}, \pi] = \text{sort}(\mathbf{h})$ ;
5    $\mathbf{r}_{\text{sort}} = \pi(\mathbf{r})$ ;
6    $\mathbf{S}_y = \mathbf{r}_{\text{sort}}\mathbf{H}^T$ ;
7    $lc = 1$ ; // loop counter initialization
8   while ( $\text{sum}(\mathbf{S}_y) > 0$ ) & ( $lc \leq n_{\Phi}$ ) do
9      $\mathbf{f}_{\text{pv}} = \Phi(l)$ ;
10     $\mathbf{r}_t = \mathbf{r}_{\text{sort}} \oplus \mathbf{f}_{\text{pv}}$ ;
11     $\mathbf{r}_o = \pi^{-1}(\mathbf{r}_t)$ ;
12     $\mathbf{S}_y = \mathbf{r}_o\mathbf{H}^T$ ;
13     $lc = lc + 1$ ;
14  end
15 else
16    $\mathbf{r}_o = \mathbf{r}$ 
17 end
18 if ( $\text{sum}(\mathbf{S}_y) == 0$ ) then
19    $\mathbf{c}_o^{\text{opt}} = \mathbf{r}_o$ ;
20 else
21    $\mathbf{c}_o^{\text{opt}} = \mathbf{r}$ ;
22 end

```

#### A. Determination of $P_{e_v}, 1 \leq v \leq d$ :

To compute  $P_{e_v}$ , we arrange the  $\rho_i$  values in ascending order, that is,  $\rho_1 \leq \rho_2 \leq \dots \leq \rho_n$ . We introduce the notation  $q_i$  as follows:

$$q_i = \text{Prob}\{n_i \geq \rho_i | \rho_i\} = Q\left(\sqrt{2\rho_i}\right) \leq e^{-\rho_i},$$

where the final inequality is derived with reference to the Chernoff bound [87]. Let  $\bar{q}_i = \mathbf{E}_{\rho_i}[q_i]$ . This implies

$$\bar{q}_1 \geq \bar{q}_2 \geq \dots \geq \bar{q}_n, \quad (4)$$

$$(1 - \bar{q}_1) \leq (1 - \bar{q}_2) \leq \dots \leq (1 - \bar{q}_n). \quad (5)$$

Initially, we derive the probability of error expressions for both single and double error scenarios. Subsequently, we generalize these expressions to encompass all  $v$  errors, where  $1 \leq v \leq d$ .

### 1) $P_{e_1}$ derivation

In this scenario, we consider a single error occurring exclusively in the MRB section of  $\mathbf{r}$ , i.e. without any errors in the LRB section. The probability of a single error occurring in the  $i$ -th location is represented by  $\bar{q}_i \prod_{j=1, j \neq i}^n (1 - \bar{q}_j)$ . When we sum over all potential locations within the MRB, we obtain:

$$P_{e_1} = \sum_{i=(d+1)}^n \bar{q}_i \prod_{\substack{j=1 \\ j \neq i}}^n (1 - \bar{q}_j).$$

Then,  $P_{e_1}$  can be upper-bounded as

$$\begin{aligned} P_{e_1} &\leq \sum_{i=(d+1)}^n \bar{q}_i (1 - \bar{q}_n)^{n-1}, \text{ (using (5))} \\ &\leq (n - d) \bar{q}_{d+1} (1 - \bar{q}_n)^{n-1}. \text{ (using (4)).} \end{aligned} \quad (6)$$

### 2) $P_{e_2}$ derivation

This scenario involves the presence of two errors. Among these two errors, at least one resides in the MRB section, which results in decoding failure. Therefore,  $P_{e_2}$  comprises two probabilities: one representing the probability of a single error occurring in the MRB ( $P_{e_{21}}$ ), and the other indicating the probability of both errors being in the MRB ( $P_{e_{22}}$ ) yielding:

$$P_{e_2} = P_{e_{21}} + P_{e_{22}}. \quad (7)$$

To calculate  $P_{e_{21}}$ , we note that the probability of two errors occurring in the  $i$ -th and  $j$ -th locations is represented as  $\bar{q}_i \bar{q}_j \prod_{k=1, k \neq i, j}^n (1 - \bar{q}_k)$ . By summing over all feasible locations within the MRB section for  $j$  and the LRB section for  $i$ , we obtain:

$$P_{e_{21}} = \sum_{i=1}^d \sum_{j=(d+1)}^n \bar{q}_i \bar{q}_j \prod_{\substack{k=1 \\ k \neq i, j}}^n (1 - \bar{q}_k).$$

We now establish an upper bound for the probability  $P_{e_{21}}$  by employing (4) and (5), as follows:

$$\begin{aligned} P_{e_{21}} &\leq \sum_{i=1}^d \sum_{j=(d+1)}^n \bar{q}_i \bar{q}_j (1 - \bar{q}_n)^{n-2}, \text{ (using (5))} \\ &= \left( \sum_{i=1}^d \bar{q}_i \right) \left[ \sum_{j=d+1}^n \bar{q}_j \right] (1 - \bar{q}_n)^{n-2}, \\ &\leq (d \bar{q}_1) [(n - d) \bar{q}_{d+1}] (1 - \bar{q}_n)^{n-2}, \text{ (using (4))} \\ &\leq d(n - d) \bar{q}_1 \bar{q}_{d+1} (1 - \bar{q}_n)^{n-2}. \end{aligned} \quad (8)$$

In a similar fashion, we formulate  $P_{e_{22}}$  as follows:

$$\begin{aligned} P_{e_{22}} &= \text{Prob}\{2 \text{ errors in MRB}\}, \\ &= \sum_{j=d+1}^{n-1} \sum_{k=j+1}^n \bar{q}_j \bar{q}_k \prod_{\substack{l=1 \\ l \neq j, k}}^n (1 - \bar{q}_l), \\ &\leq \sum_{j=d+1}^{n-1} (n - j) \bar{q}_{d+1}^2 (1 - \bar{q}_n)^{n-2} \text{ (using (4), (5))} \\ &\leq \left( \sum_{j=d+1}^{n-1} n - \sum_{j=d+1}^{n-1} j \bar{q}_{d+1}^2 \right) (1 - \bar{q}_n)^{n-2} \\ &\leq \binom{n-2}{2} \bar{q}_{d+1}^2 (1 - \bar{q}_n)^{n-2}. \end{aligned} \quad (9)$$

### 3) $P_{e_v}$ ( $1 \leq v \leq d$ ) derivation

Building upon the insights gained from the preceding two subsections, where we calculated the error probabilities for the single and double error cases, we extend our analysis to derive the error probability for scenarios involving  $v$  errors. Among these  $v$  errors, we denote the number of errors in the MRB section by  $v_m$  and the number of errors in the LRB section by  $v_n$ . Hence, we have:

$$v = v_m + v_n. \quad (10)$$

We denote  $P_{e_v}(v_m, v_n)$  as the probability associated with  $v_m$  errors in the MRB section and the  $v_n$  errors in the LRB section. Consequently,  $P_{e_v}$  can be expressed as follows:

$$P_{e_v} = \sum_{v_m=1}^v P_{e_v}(v_m, v_n). \quad (11)$$

Let us now derive  $P_{e_v}(v_m, v_n)$  as follows:

$$\begin{aligned} P_{e_v}(v_m, v_n) &= \left( \sum_{j_1=d+1}^{n-v_m+1} \sum_{j_2=j_1+1}^{n-v_m} \cdots \sum_{j_{v_m}=j_{(v_m-1)+1}}^n \bar{q}_{j_1} \bar{q}_{j_2} \cdots \bar{q}_{j_{v_m}} \right) \\ &\quad \left( \sum_{k_1=1}^d \sum_{k_2=k_1+1}^{d-1} \cdots \sum_{k_{v_n}=k_{(v_n-1)+1}}^{d-v_n+1} \bar{q}_{k_1} \bar{q}_{k_2} \cdots \bar{q}_{k_{v_n}} \right) \\ &\quad \left( \prod_{\substack{v=1 \\ v \neq j_1, j_2, \dots, j_{v_m} \\ v \neq k_1, k_2, \dots, k_{v_n}}}^n (1 - \bar{q}_v) \right), \\ &\leq \left( \binom{n-d}{v_m} \bar{q}_{d+1} \bar{q}_{d+2} \cdots \bar{q}_{d+v_m} \right) \\ &\quad \left( \binom{d}{v_n} \bar{q}_1 \bar{q}_2 \cdots \bar{q}_{v_n} \right) ((1 - \bar{q}_n)^{n-v}), \text{ (using (5))} \\ &\leq \left( \binom{n-d}{v_m} (\bar{q}_{d+1})^{v_m} \right) \\ &\quad \left( \binom{d}{v_n} (\bar{q}_1) \right)^{v_n} ((1 - \bar{q}_n)^{n-v}), \text{ (using (4))} \\ &\leq \binom{n-d}{v_m} \binom{d}{v_n} (\bar{q}_{d+1})^{v_m} (\bar{q}_1)^{v_n} (1 - \bar{q}_n)^{n-v}. \end{aligned} \quad (12)$$

Upon substituting (12) into (11), we arrive at:

$$P_{e_v} \leq \sum_{v_m=1}^v \binom{n-d}{v_m} \binom{d}{v_n} (\bar{q}_{d+1})^{v_m} (\bar{q}_1)^{v_n} (1 - \bar{q}_n)^{n-v}. \quad (13)$$

#### 4) Derivation of $\bar{q}_1$ , $\bar{q}_{d+1}$ and $\bar{q}_n$

Observe that the upper bound in (13) relies on the values of  $\bar{q}_1$ ,  $\bar{q}_{d+1}$ , and  $\bar{q}_n$ . To derive expressions for these values, we must delve into the associated ordered statistics.

Consider a continuous random variable  $Y$  having a probability density function (PDF)  $f(Y)$  and cumulative distribution function (CDF)  $F(Y)$ . Let  $(Y_1, Y_2, \dots, Y_n)$  represent a random sample, and  $(Y_{(1)}, Y_{(2)}, \dots, Y_{(n)})$  denote the corresponding sample sorted in ascending order so that  $Y_{(1)} < Y_{(2)} < \dots < Y_{(n)}$ . The PDF of the  $r^{th}$  order statistics,  $Y_{(r)}$  [88], can be expressed as:

$$f_{Y_{(r)}} = \frac{n!}{(r-1)!(n-r)!} [F(y)]^{r-1} [1-F(y)]^{n-r} f(y). \quad (14)$$

Let us now define  $a = \frac{n!}{(r-1)!(n-r)!}$ , and let  $Y_i$  be represented by  $\rho_i$ , where a gain  $\rho_i = h_i^2 \bar{\rho}_c$  denotes the instantaneous SNR and follows an exponential distribution. Upon substituting the corresponding CDF  $F(Y)$  and PDF  $f(y)$  into (14), we obtain:

$$\begin{aligned} f_{Y_{(r)}} &= a \left(1 - e^{-\frac{y}{\bar{\rho}_c}}\right)^{r-1} \left(e^{-\frac{y}{\bar{\rho}_c}}\right)^{n-r} \left(\frac{1}{\bar{\rho}_c} e^{-\frac{y}{\bar{\rho}_c}}\right), \\ &= \frac{a}{\bar{\rho}_c} \left(e^{-\frac{y}{\bar{\rho}_c}}\right)^{n-r+1} \left(1 - e^{-\frac{y}{\bar{\rho}_c}}\right)^{r-1}, \\ &= \frac{a}{\bar{\rho}_c} \sum_{m=0}^{r-1} \binom{r-1}{m} (-1)^m e^{-\frac{my}{\bar{\rho}_c}} e^{-\frac{y(n-r+1)}{\bar{\rho}_c}}, \\ &= \frac{a}{\bar{\rho}_c} \sum_{m=0}^{r-1} \binom{r-1}{m} (-1)^m e^{-\frac{y}{\bar{\rho}_c} (n-r+1+m)}. \end{aligned} \quad (15)$$

To determine  $\bar{q}_r$ , we calculate its expectation using (15). This leads us to the following expression:

$$\begin{aligned} \bar{q}_r &= \int_0^\infty \frac{a}{\bar{\rho}_c} \sum_{m=0}^{r-1} \binom{r-1}{m} (-1)^m e^{-\frac{y}{\bar{\rho}_c} (n-r+1+m)} e^{-y} dy, \\ &= \int_0^\infty \frac{a}{\bar{\rho}_c} \sum_{m=0}^{r-1} \binom{r-1}{m} (-1)^m e^{-\frac{y}{\bar{\rho}_c} (n-r+1+m+\bar{\rho}_c)}, \\ &= a \left( \sum_{m=0}^{r-1} \binom{r-1}{m} \frac{(-1)^m}{n-r+1+m+\bar{\rho}_c} \right), \\ &= \frac{a(r-1)!}{\prod_{k=0}^{r-1} (\bar{\rho}_c + n - k)}, \\ &= \frac{(n!/(n-r)!)}{\prod_{k=0}^{r-1} (\bar{\rho}_c + n - k)}, \\ &= \frac{\prod_{j=0}^{r-1} (n-j)}{\prod_{k=0}^{r-1} (\bar{\rho}_c + n - k)}, \\ &= \frac{n P_r}{\prod_{k=0}^{r-1} (\bar{\rho}_c + n - k)}. \end{aligned} \quad (16)$$

By substituting the values  $r = 1$ ,  $r = (d+1)$ , and  $r = n$  into (16), we arrive at

$$\bar{q}_1 = \frac{n}{n + \bar{\rho}_c}. \quad (17)$$

$$\bar{q}_{d+1} = \frac{n P_{d+1}}{\prod_{k=0}^d (\bar{\rho}_c + n - k)}. \quad (18)$$

$$\bar{q}_n = \frac{1}{\prod_{k=0}^{n-1} (\bar{\rho}_c + n - k)}. \quad (19)$$

#### 5) Derivation of $P_{e_v} (1 \leq v \leq d)$

By substituting equations (17), (18), and (19) into (13), we arrive at:

$$P_{e_v} \leq \sum_{v_m=1}^l \binom{n-d}{v_m} \binom{d}{v_n} \left( \frac{n P_{d+1}}{\prod_{k=0}^d (\bar{\rho}_c + n - k)} \right)^{v_m} \left( \frac{n}{n + \bar{\rho}_c} \right)^{v_n} \left( 1 - \frac{1}{\prod_{k=0}^{n-1} (\bar{\rho}_c + n - k)} \right)^{n-v}. \quad (20)$$

#### B. Derivation of $P_{e_{(d+1)}}$ :

This scenario arises when the number of errors in the received word exceeds the limit of  $d = (d_{\min} - 1)$ . The proposed decoder is capable of correcting up to  $d$  errors since it allows a maximum of  $d = (d_{\min} - 1)$  flips in the decoding process. In such cases, the  $\rho_i$  values are unordered, and the probability of encountering  $d_{\min}$  or more errors can be formulated as

$$P_{e_{(d+1)}} = \sum_{m=d_{\min}}^n \binom{n}{m} \bar{q}^m (1 - \bar{q})^{n-m}. \quad (21)$$

Here,  $\bar{q}$  represents the average probability of bit error associated with the unordered  $\rho_i$  values. In terms of the average SNR,  $\bar{q}$  can be expressed as:

$$\bar{q} = \mathbf{E}_\rho [Prob\{n \geq \rho | \rho\}] = \mathbf{E}_\rho \left[ Q \left( \sqrt{2\rho} \right) \right] \leq \frac{1}{1 + \bar{\rho}_c}. \quad (22)$$

We now express the upper bound on  $\bar{q}$  by applying the Chernoff bound [87], followed by averaging over the distribution of  $\rho$  upon substituting (22) into (21), we arrive at:

$$P_{e_{(d+1)}} = \sum_{m=(d+1)}^n \binom{n}{m} \left\{ \frac{1}{1 + \bar{\rho}_c} \right\}^m \left\{ \frac{\bar{\rho}_c}{1 + \bar{\rho}_c} \right\}^{n-m}. \quad (23)$$

We are now ready to establish an upper bound on the total average probability of error by incorporating (20) and (23) into (3), which yields (24). The upper bound shown by the equation (24) is loose bound but gives the same diversity order.

#### C. The Diversity Flip Decoder's Diversity order

Let us now proceed by determining the diversity order of the DFD.

#### Theorem 1:

$$P_{FD}(\mathcal{C}) \leq \sum_{v=1}^d \sum_{v_m=1}^v \binom{n-d}{v_m} \binom{d}{v_n} \left( \frac{{}^nP_{d+1}}{\prod_{k=0}^d (\bar{\rho}_c + n - k)} \right)^{v_m} \left( \frac{n}{n + \bar{\rho}_c} \right)^{v_n} \left( 1 - \frac{1}{\prod_{k=0}^{n-1} (\bar{\rho}_c + n - k)} \right)^{n-v} + \sum_{m=(d+1)}^n \binom{n}{m} \left\{ \frac{1}{1 + \bar{\rho}_c} \right\}^m \left\{ \frac{\bar{\rho}_c}{1 + \bar{\rho}_c} \right\}^{n-m} \quad (24)$$

The DFD achieves a diversity order of  $d_{\min}$  for a binary linear block code  $\mathcal{C}(n, k)$  having a minimum Hamming distance of  $d_{\min}$ . ■

*Proof:*

Referring to an equation (24), let

$$\mathcal{A} = \sum_{v=1}^d \sum_{v_m=1}^v \binom{n-d}{v_m} \binom{d}{v_n} \left( \frac{{}^nP_{d+1}}{\prod_{k=0}^d (\bar{\rho}_c + n - k)} \right)^{v_m} \left( \frac{n}{n + \bar{\rho}_c} \right)^{v_n} \left( 1 - \frac{1}{\prod_{k=0}^{n-1} (\bar{\rho}_c + n - k)} \right)^{n-v} \quad (25)$$

$$\mathcal{B} = \sum_{m=(d+1)}^n \binom{n}{m} \left\{ \frac{1}{1 + \bar{\rho}_c} \right\}^m \left\{ \frac{\bar{\rho}_c}{1 + \bar{\rho}_c} \right\}^{n-m}$$

$$\mathcal{B} = \left[ \binom{n}{d+1} \left\{ \frac{1}{1 + \bar{\rho}_c} \right\}^{(d+1)} \left\{ \frac{\bar{\rho}_c}{1 + \bar{\rho}_c} \right\}^{n-(d+1)} \right] +$$

$$\left[ \binom{n}{d+2} \left\{ \frac{1}{1 + \bar{\rho}_c} \right\}^{(d+2)} \left\{ \frac{\bar{\rho}_c}{1 + \bar{\rho}_c} \right\}^{n-(d+2)} \right] + \dots$$

$$\dots + \left[ \binom{n}{n} \left\{ \frac{1}{1 + \bar{\rho}_c} \right\}^n \right]. \quad (26)$$

Then, Equation (24) can be written as

$$P_{FD}(\mathcal{C}) = \mathcal{A} + \mathcal{B}. \quad (27)$$

By invoking the definition of diversity order as outlined in [23] in the limit (27) can be expressed as

$$D = \lim_{\bar{\rho}_c \rightarrow \infty} \frac{-\log[P_{FD}(\mathcal{C})]}{\log[\bar{\rho}_c]}$$

$$= \lim_{\bar{\rho}_c \rightarrow \infty} \frac{-\log[\mathcal{A} + \mathcal{B}]}{\log[\bar{\rho}_c]}.$$

But, as  $\bar{\rho}_c \rightarrow \infty$ , the term  $\mathcal{A} \rightarrow 0$ , hence:

$$D = \lim_{\bar{\rho}_c \rightarrow \infty} \frac{-\log[\mathcal{B}]}{\log[\bar{\rho}_c]}.$$

As  $\bar{\rho}_c \rightarrow \infty$ , the exponent  $(d+1)$  starts to dominate over other exponentials in the term B (equation (26)). Therefore we have

$$D = d + 1$$

$$= d_{\min}. \quad (28)$$

#### D. Computational Complexity Analysis

The DFD combines the benefits of low complexity with the preservation of diversity. It relies on a comprehensive list of flip vectors denoted as  $\mathbf{f}_{\text{pv}}$ , which encompasses a total of  $2^d - 1$  ( $2^{d_{\min}-1} - 1$ ) vectors. The computational complexity of the DFD is directly influenced by the minimum distance ( $d_{\min}$ ) of the code  $\mathcal{C}(n, k)$ . Through a systematic search process within the list  $\Phi$ , the algorithm aims for identifying a suitable flip vector by sequentially evaluating each vector. Even in the worst-case scenario, where Algorithm 1 examines the last flip vector in  $\Phi$ , the DFD maintains its efficiency, given its worst-case complexity order of  $2^{d_{\min}-1} - 1$ . Therefore, this technique has the twin benefits of low computational complexity, while preserving diversity.

Table III provides a detailed comparison of the maximum complexities associated with various decoders.

TABLE III: Decoder Complexity: A Comparative Study

Decoder	Maximum complexity
Chase-1	$\left( \lfloor \frac{d_{\min}}{2} \rfloor \right) \times \text{binary decoder complexity}$
Chase-2	$2^{\lfloor \frac{d_{\min}}{2} \rfloor} \times \text{binary decoder complexity}$
Chase-3	$\lfloor \frac{d_{\min}}{2} \rfloor + 1 \times \text{binary decoder complexity}$
OSD	$\mathcal{O}(k^l)$
ISD	$\mathcal{O}(k^l)$
GRAND	$b$ (Threshold)
ORBGAND	$b$ (Threshold)
Proposed DFD	$2^{d_{\min}-1} - 1$

#### V. Simulation Results - DFD

To evaluate the performance of the proposed DFD, several examples are considered. The simulations are conducted in an uncorrelated Rayleigh fading channel with unit variance, using BPSK modulation throughout.

To illustrate the effectiveness of the DFD technique, Bit Error Rate (BER) vs signal-to-noise ratio per message bit ( $E_b/N_0$ ) plots are generated. It should be noted that  $\bar{\rho}_c$  and  $E_b/N_0$  are related by  $E_b/N_0 = \bar{\rho}_c \times \text{code rate}$ . To achieve a desired BER on the order of  $10^{-a}$  ( $a \in \mathbb{Z}^+$ ), a transmission of  $10^{(a+2)}$  bits is performed for the BER simulation plots.

The subsequent section showcases the versatile capability of the proposed Diversity Flip decoder (DFD) in achieving

diversity at a low complexity for various code types, thereby establishing its universality. Additionally, we conduct a thorough comparison of the diversity flip decoder's performance vs complexity against those of existing decoding methods in the following subsection.

#### A. Simulations: DFD Preserves Diversity

In order to illustrate the performance and diversity preservation capability of the proposed DFD, the simulation results are presented for four specific codes: Parity check codes, Hamming codes, BCH codes, and Polar codes. However, it is important to emphasize that the DFD applies to all types of linear codes.

- **Example-1:-Parity check codes:** Our initial evaluation is focused on Parity check (PC) codes, which exhibit a minimum distance of 2. When a parity mismatch arises at the receiver, the DFD technique acts by toggling the LRB bit associated with the lowest Channel State Information (CSI). Table IV provides the detailed parameters for the PC codes employed in the simulations.

Figure 2 provides the BER curves of the PC (4, 3) and PC (12, 11) as representative examples, which benchmarked both against HDD (hard-decision decoding) and SDD (soft-decision decoding). At high SNR levels, the DFD curve mirrors the slope of the SDD curve, indicating an equivalent level of diversity capability to that of soft-decision decoding. This observation clearly demonstrates that, as shown in Figure 2, the DFD technique achieves a diversity order of 2 for PC codes operating in a Rayleigh fading channel. Furthermore, Figure 3 extends our performance evaluations of DFD in comparison to hard-decision decoding for the PC (64, 63), (128, 127), and (256, 255) codes. These results indicate that DFD attains a second-order diversity for PC codes.

It is worth noting that PC codes possess a minimum distance of  $d_{min} = 2$ , thus lacking error-correcting capabilities ( $t = 0$ ). As a result, PC codes are primarily employed for error detection purposes. However, the proposed DFD technique exhibits the capability of correcting single errors with high probability in the presence of fading, despite having  $d_{min} = 2$ .

- **Example-2:- Hamming codes:** Next, we examine the performance of the DFD technique for Hamming codes, which offer a minimum distance of 3. In this case, two bits associated with the two lowest CSI values of the received word are selected for flipping, as per Algorithm 1. There are three possible ways of flipping these two bits, until a valid codeword is obtained. The parameters of the Hamming codes used for the simulations are provided in Table V.

Figure 4 presents a comparison of the DFD to HDD and SDD for the (7, 4) and (15, 11) Hamming codes. At high Signal-to-Noise Ratio (SNR), both the DFD curve and the SDD curve exhibit the same slope for

both the (7, 4) and (15, 11) Hamming codes, indicating a diversity order of 3, which is equivalent to that of soft-decision decoding. Additionally, the figure displays the curves for hard-decision decoding and the DFD used for the (127, 120) and (255, 247) Hamming codes, further confirming that the DFD technique achieves a diversity order of 3 for Hamming codes.

- **Example-3:- BCH codes:** We now turn our attention to BCH codes. The simulation parameters for the corresponding BCH codes are presented in Table VI.

Figure 5 presents the performance of the DFD technique applied to BCH codes. The plot offers a comparative analysis involving the DFD technique, HDD, and SDD for the (15, 7) BCH code. At high SNR values, the DFD mirrors the slope of the SDD curve, demonstrating a diversity order equivalent to  $d_{min}$  (minimum distance). Furthermore, the figure extends the evaluation by including the curves of the DFD and of the hard decision aided (255, 239), (63, 51), and (31, 21) BCH codes, conclusively confirming that the proposed DFD attains a diversity order of  $d_{min}$ .

It is worth highlighting that the IEEE 802.15.6 standard of Wireless Body Area Networks (WBAN) [89] has chosen to employ short BCH codes, specifically those with block lengths of  $n = 63$  and  $n = 31$ . The proposed DFD outperforms the classic Peterson algorithm in the Rayleigh fading channel while preserving diversity. Moreover, it is important to emphasize that the DFD technique demonstrates reduced computational complexity in comparison to the Peterson algorithm, a point that will be elaborated on in the following discussion. Peterson's algorithm and the Berlekamp-Massey algorithm (BM) are commonly utilized for the decoding of BCH codes. Peterson's algorithm exhibits a worst-case complexity of  $2t(n-1)^2 + \Theta(t^3)$ , whereas the Berlekamp-Massey algorithm (BM) has a worst-case complexity of  $2t(n-1)^2 + \Theta(t^2)$  [90]. By contrast, the worst-case complexity of the proposed DFD, expressed as  $2^{(d_{min}-1)} - 1$ , is substantially lower.

- **Example-4:- Polar codes:** Finally, the family of Polar codes was examined to evaluate the performance of the DFD. Various Polar codes were selected for simulation, and the simulation parameters are provided in Table VII. The minimum distance ( $d_{min}$ ) of the Polar codes was determined using the *gfweight()* function in MATLAB. The BER performance using the proposed DFD is illustrated in Figure 6. Interestingly, the Polar codes (128, 113), (128, 106), and (128, 99) exhibit the same minimum distance of  $d_{min} = 4$ . Furthermore, the BER curves for these codes exhibit identical slopes, indicating the same diversity of  $d_{min} = 4$ .

It is worth noting that the performance of the original GRAND technique [53] aligns with that of HDD. By contrast, the proposed DFD outperforms the original GRAND



TABLE IV: Parameters of PC Codes

$(n, k)$	$d_{\min}$	Correcting capability, $t$	Maximum No. of Correctable Errors with FD
(4,3)	2	0	1
(12,11)	2	0	1
(64,63)	2	0	1
(128,127)	2	0	1
(256,255)	2	0	1

TABLE V: Parameters of Hamming Codes

$(n, k)$	$d_{\min}$	Correcting capability, $t$	Maximum No. of Correctable Errors with FD
(7,4)	3	1	2
(15,11)	3	1	2
(127,120)	3	1	2
(255,247)	3	1	2

at high SNRs, while maintaining exceptionally low computational complexity, as seen later in Figure 9.

### B. Simulations: Performance analysis of DFD

To assess the performance of the proposed DFD, the coding gains attained at BER of  $10^{-6}$  are tabulated in Tables VIII and IX for different code lengths and code rates in case of Hamming and BCH codes, respectively. Figure 7 shows the plot of coding gain versus code rate in the case of both

TABLE VI: Parameters of BCH codes

$(n, k)$	$d_{\min}$	Correcting capability, $t$	Maximum No. of Correctable Errors with FD
(15,7)	5	2	4
(31,21)	5	2	4
(63,51)	5	2	4
(255,239)	5	2	4

TABLE VII: Parameters of Polar Codes

$(n, k)$	$d_{\min}$	Maximum No. of Correctable Errors with FD
(128,120)	2	1
(128,113)	4	3
(128,106)	4	3
(128,99)	4	3

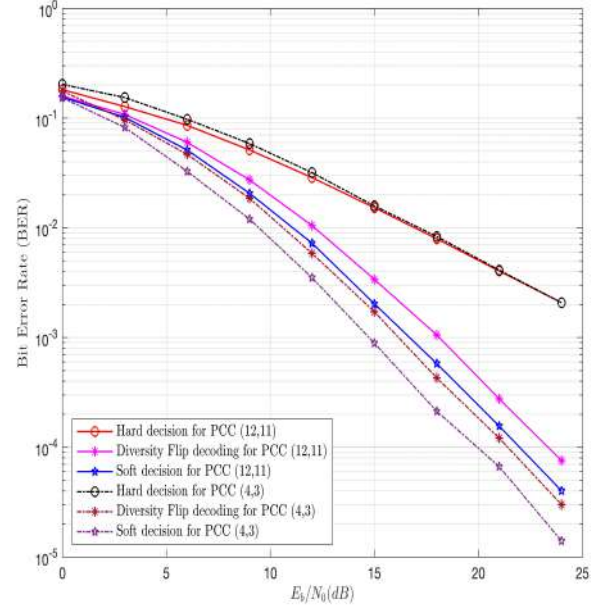


FIGURE 2: BER of various short PC codes using the proposed DFD.

TABLE VIII: Coding gain offered by DFD in case of Hamming codes over uncorrelated Rayleigh fading channel

$n$	$k$	Code Rate (R)	$E_b/N_0$ at BER of		Coding Gain at BER of	
			$10^{-5}$	$10^{-6}$	$10^{-5}$	$10^{-6}$
7	4	0.57	19dB	22.2dB	24.9dB	31.8dB
15	11	0.73	20.2dB	23.6dB	23.7dB	30.4dB
31	26	0.83	21.7dB	25.2dB	22.2dB	28.8dB
63	57	0.9	23.4dB	26.8dB	20.5dB	27.2dB
127	120	0.94	25dB	28.6dB	18.9dB	25.4dB

BCH and Hamming codes for different code lengths and code rates.

### C. Simulation-Based Comparative Analysis: Performance versus Complexity of Proposed DFD versus Existing Techniques

In order to assess the performance versus complexity of the proposed diversity flip decoder in comparison to existing techniques, the BCH (127,113) code was selected as a case study. Figure 8 showcases the BER performance comparison between the DFD decoder and several existing approaches, such as hard-decision GRAND [54], soft-decision ORBGRAND [57], ORBGRAND with pseudo-soft-information [61], and fading-GRAND [60]. In all the

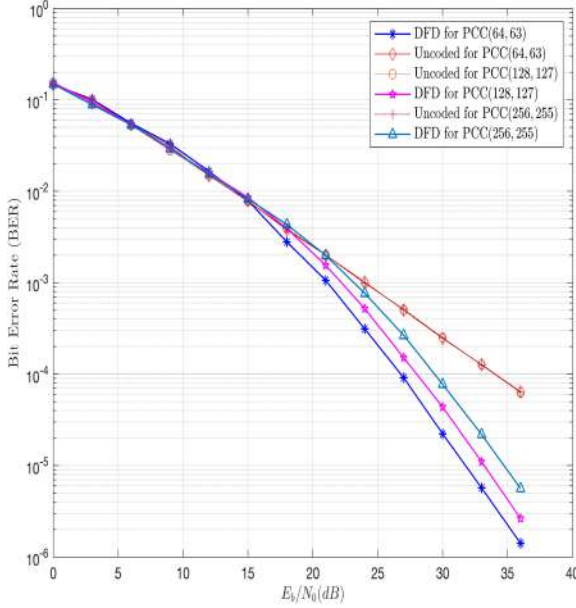


FIGURE 3: BER of various long PC codes using the proposed DFD.

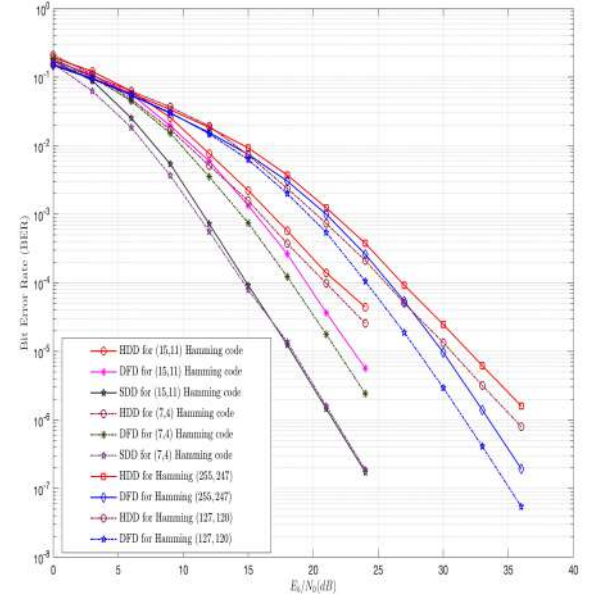


FIGURE 4: BER of various Hamming codes using the proposed DFD.

simulations conducted involving GRAND and its variants, the Abandonment Bit value (ABB) is set to  $10^6$ . In the context of fading-GRAND, the threshold  $\Delta$  is determined by assigning  $m = -0.02165$  and  $b = 0.7924$ , as stated in [60]. Observe in Figure 8 that for  $E_b/N_o > 20$  dB the proposed DFD performs better than the original hard-decision GRAND.

The GRAND and its variants such as ORBGRAND [58], fading-GRAND [60], and ORBGRAND [61] associated with pseudo-soft-information, have the following pair of dominant complexity contributions. Firstly, the generation of subsequent tentative noise sequences contributes to the overall complexity. Secondly, the number of codebook accesses required for validating the codeword obtained or calculating the syndrome equation adds to the complexity. By contrast, DFD does not have to generate the next flip vector, since the set of flip vectors is already available at the receiver for a given minimum distance  $d_{min}$  and codeword length  $n$ . The DFD only involves syndrome calculations or codebook queries, hence imposing a low complexity. Therefore, to facilitate a comparison between the DFD and other methods, the number of codebook queries is taken into consideration for complexity comparison throughout this paper. However, the actual overall complexity of GRAND and its variants exceeds the complexity shown in the results of this paper.

The complexity of the decoding algorithms varies depending on both the number and location of errors in the received codeword. Consequently, the overall complexity of decoders exhibits variations versus the SNR values. Since the

worst-case complexity determines the maximum hardware-dimension and power-consumption in system design, we present the worst-case complexities in Figure 9. The results clearly demonstrate that the proposed DFD exhibits significantly lower worst-case complexities than the alternative methods.

Figure 10 illustrates the coding gain at a bit error rate (BER) of  $10^{-6}$  against worst-case complexity for the DFD, GRAND and its variants. Observe that the DFD achieves an approximately 2 dB higher coding gain than the original hard-decision GRAND at 4 orders lower complexity.

To underscore the minimal complexity of the DFD in comparison to alternative counterparts, Table X tabulates the percentage complexities of the various decoders. The complexities are computed relative to that of the GRAND [53] at BER of  $10^{-5}$ . Regardless, of whether we consider the maximum (worst-case) complexity or average complexity, the DFD exhibits significantly lower complexity than the other decoders.

The GRAND decoder, known for its hard-decision decoding approach, is outperformed by the proposed hard-decision DFD, which exhibits superior performance at a modest computational complexity. Although ORBGRAND (soft-decision decoder), fading-GRAND, and ORBGRAND relying on pseudo-soft information provide improved performance compared to the DFD technique, the DFD has the edge in terms of low complexity. Hence the DFD is an attractive choice for light-weight real-world applications.

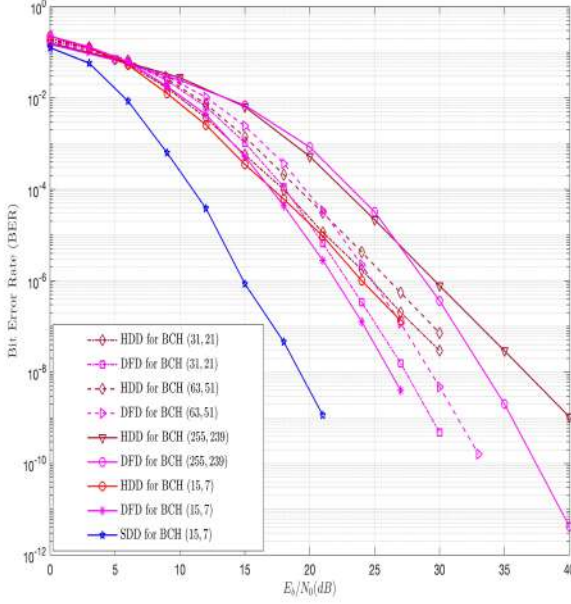


FIGURE 5: BER of various BCH codes using the proposed DFD.

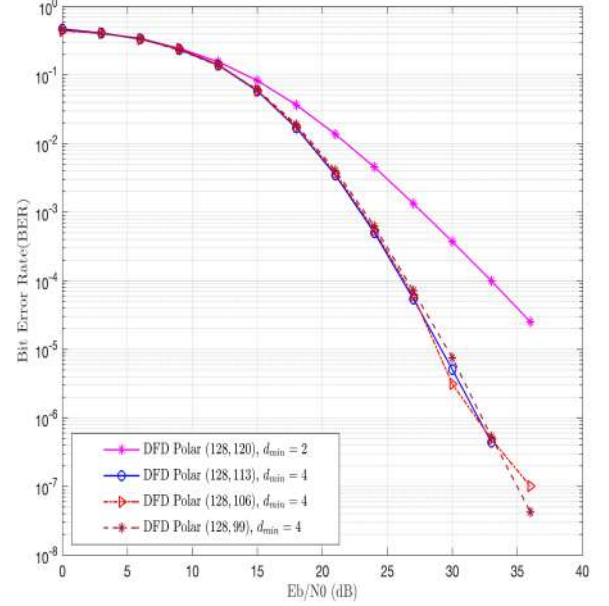


FIGURE 6: BER of various Polar codes using the proposed DFD.

## VI. Extended Diversity Flip Decoder

The Diversity Flip decoder, proposed in Section III flips minimally  $d_{min} - 1$  least reliable bits in diverse configurations, thereby maintaining the diversity order across fading channels. In this section, we introduce an Extended Diversity Flip decoder (EDFD), which extends the window of the least reliable bits to be flipped. Moreover, in EDFD, the flip vector corresponding to a valid codeword having the minimum Hamming weight is chosen as the final error vector for obtaining the decoded codeword.

Let  $\epsilon$  be the extension window length of the EDFD and denote the flip vector by  $\mathbf{fe}_{pv}^n$ . Further more, let the set of flip vectors be represented by  $\Phi_e$ , which is defined as follows:

$$\Phi_e = \left\{ \mathbf{fe}_{pv}^n \mid \mathbf{fe}_{pv}^n = \begin{bmatrix} \mathbf{fep}_i^{d+\epsilon} & \mathbf{0}^{n-d-\epsilon} \end{bmatrix} \right\} \quad (1 \leq i \leq d).$$

In this context,  $\mathbf{fep}_i^{d+\epsilon}$  represents the flip pattern of the EDFD, characterized by a length of  $d + \epsilon$  and a Hamming weight of  $i$ . For instance, for a (7, 4, 3) Hamming code, we have  $d_{min} = 3$ , thus  $d = d_{min} - 1 = 2$ . Subsequently, the flip vector set, denoted as  $\Phi_e$ , associated with  $\epsilon = 1$  is

elucidated below:

$$\Phi = \left\{ \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \right. \\ \left. \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \right. \\ \left. \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}, \right. \\ \left. \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \right. \\ \left. \begin{bmatrix} 1 & 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}, \right. \\ \left. \begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}, \right. \\ \left. \right\}$$

The difference between DFD and EDFD is in their respective flip vector lists. The decoding procedure of EDFD mirrors that of DFD, as detailed below. Initially, the Channel State Information (CSI), denoted as  $\mathbf{h}$  is subjected to sorting to yield  $\mathbf{h}_{sort}$ , alongside its corresponding permutation function,  $\pi_1$ . Subsequently,  $\pi_1$  is applied to the received codeword  $\mathbf{r}$  to obtain  $\mathbf{r}_{sort}$ . The first  $(d + \epsilon)$  bits within  $\mathbf{r}_{sort}$  represent the Least Reliable Bits (LRB), while the remaining bits constitute the Most Reliable Bits (MRB). Following this, the initial flip vector from the set  $\Psi_e$  is selected and added to  $\mathbf{r}_{sort}$ , yielding  $\mathbf{r}_t$ . The inverse of  $\pi_1$ , denoted as  $\pi_1^{-1}$ , is then applied to  $\mathbf{r}_t$  to recover  $\mathbf{r}_o$ . Subsequently, the syndrome for  $\mathbf{r}_o$  is computed to ascertain the codeword's validity. If a non-zero syndrome is obtained, the process iterates by selecting the subsequent flip vector from the  $\Psi_e$  list. This iteration is continued until a zero syndrome is arrived at. Upon obtaining a zero syndrome, the process terminates,

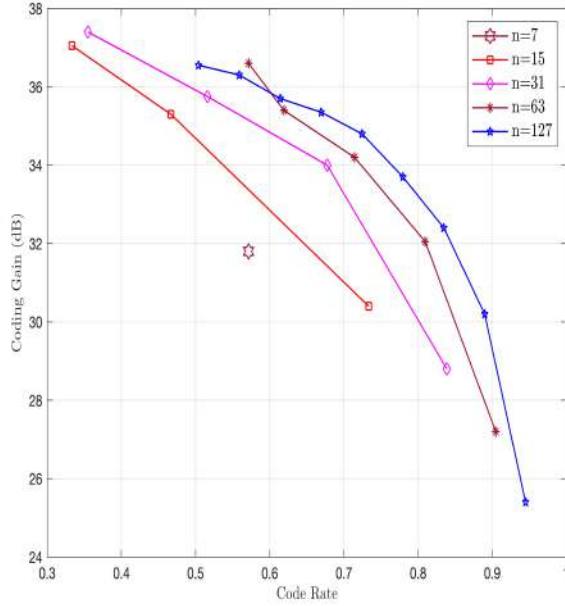


FIGURE 7: Coding Gain of DFD at BER of  $10^{-6}$  for different code lengths and code rates in case of Hamming and BCH codes in uncorrelated Rayleigh fading channel.

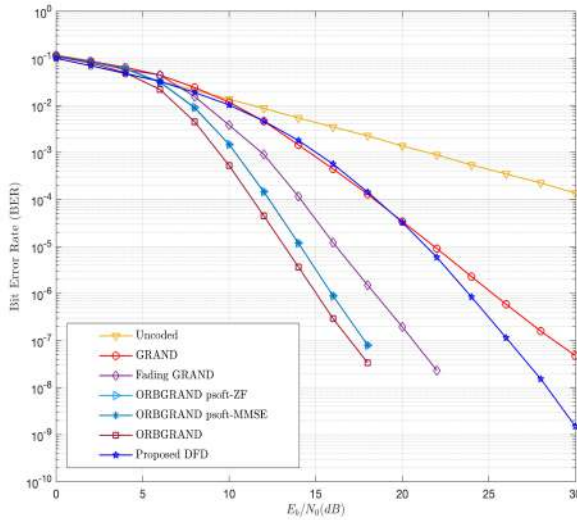


FIGURE 8: BER Comparison of the proposed DFD to existing decoders:  $BCH(127, 113)$ .

with  $\mathbf{r}_o$  representing the decoded codeword. The EDFD technique is formally summarized in Algorithm 2.

The flip vector patterns of EDFD for the (15,7,5) BCH code when  $\epsilon = 0, 1, 2, 3$  are depicted in Figures 11, 12, 13, and 14 correspondingly.

TABLE IX: Coding gain offered by DFD in case of BCH codes over uncorrelated Rayleigh fading channel

$n$	$k$	Code Rate (R)	$E_b/N_0$ at BER of		Coding Gain at BER of	
			$10^{-5}$	$10^{-6}$	$10^{-5}$	$10^{-6}$
15	7	0.46	16.1dB	18.7dB	27.8dB	35.3dB
	5	0.33	15.2dB	16.95dB	28.7dB	37.05dB
31	21	0.67	17.8dB	20dB	26.1dB	34dB
	16	0.51	16.4dB	18.25dB	27.5dB	35.75dB
	11	0.35	14.9dB	16.6dB	29dB	37.4dB
63	51	0.83	19.3dB	21.95dB	24.6dB	32.05dB
	45	0.9	17.6dB	19.8dB	26.3dB	34.2dB
	39	0.9	16.7dB	18.6dB	27.2dB	35.4dB
	36	0.9	15.7dB	17.4dB	28.2dB	36.6dB
127	113	0.89	21.4dB	23.8dB	22.5dB	30.2dB
	106	0.83	19.4dB	21.6dB	24.5dB	32.4dB
	99	0.78	18.2dB	20.3dB	25.7dB	33.7dB
	92	0.72	17.4dB	19.2dB	26.5dB	34.8dB
	85	0.67	17.05dB	18.65dB	26.85dB	35.35dB
	78	0.61	16.8dB	18.3dB	27.1dB	35.7dB
	71	0.56	16.15dB	17.7dB	27.75dB	36.3dB
	64	0.5	16.1dB	17.45dB	27.8dB	36.55dB

TABLE X: Comparative analysis:  $BCH(127, 113)$

Decoder	At BER of $10^{-5}$		
	$E_b/N_0$	Percentage of Maximum complexity with respect to GRAND	Percentage of Average complexity with respect to GRAND
GRAND [53]	22 dB	100%	100%
Fading GRAND [60]	16.2 dB	969.9321%	272.3312%
ORBGRAND psoft-ZF [61]	14.1 dB	47.895%	32.1709%
ORBGRAND psoft-MMSE [61]	14.1 dB	59.7907%	32.1709%
ORBGRAND [58]	13.2 dB	47.895%	44.4365%
<b>Proposed DFD</b>	<b>21.4 dB</b>	<b>0.0146%</b>	<b>3.6619%</b>

## VII. Simulation Results: EDFD

This section presents the simulation results for the Extended Diversity Flip Decoder (EDFD) conducted under the same

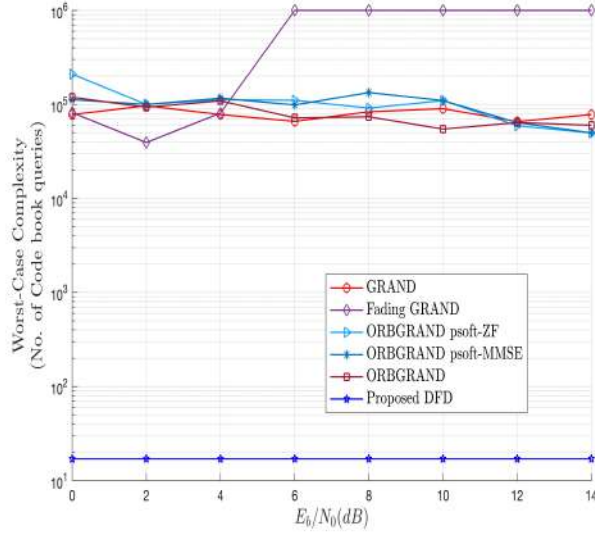


FIGURE 9: Worst-case complexity comparisons in terms of the number of codebook queries of the proposed DFD to existing decoders:  $BCH(127, 113)$ .

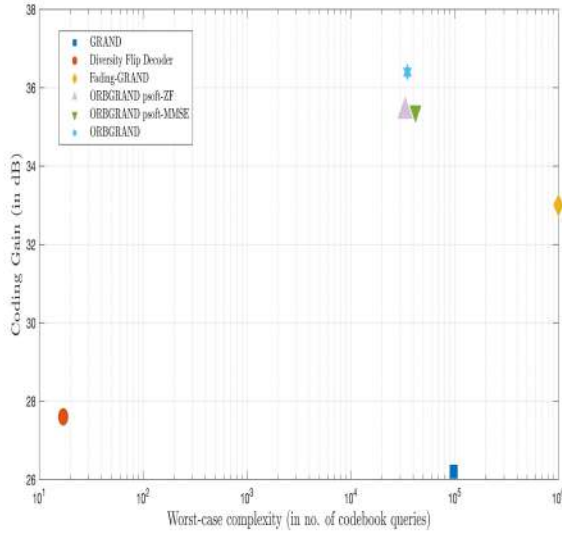


FIGURE 10: Scatter plot of coding gain at BER of  $10^{-6}$  vs. worst-case complexity:  $BCH(127, 113)$

experimental setup as that of the Diversity Flip Decoder (DFD). The simulations are conducted in a Rayleigh fading channel having unit variance, employing Binary Phase Shift Keying (BPSK) modulation. Figure 15 illustrates the Bit Error Rate (BER) simulation outcomes of the EDFD applied to a  $BCH(15, 7)$  code for varying values of the extension window length  $\epsilon$ . It is observed that as the value of  $\epsilon$  increases, there is an enhancement in the BER performance. The coding gains offered by EDFD for different values of  $\epsilon$

#### Algorithm 2: Extended Diversity Flip decoder

##### Input:

- $\Phi_e$ : Flip vector list of EDFD.
- $\mathbf{r}$ : received codeword.
- $\mathbf{h}$ : CSI vector.
- $\mathbf{H}$
- $\epsilon$ : Extension window length.
- $d_{\min}$
- $n_{\Phi_e}$ : Total number of Flip vectors of EDFD.

##### Output:

- $\mathbf{c}_o^{\text{opt}}$ : The decoded codeword.

```

1  $\mathbf{S}_y = \mathbf{r}\mathbf{H}^T$ ;
2 if ( $\text{sum}(\mathbf{S}_y) > 0$ ) then
3    $[\mathbf{h}_{\text{sort}}, \pi_1] = \text{sort}(\mathbf{h})$ ;
4    $\mathbf{r}_{\text{sort}} = \pi_1(\mathbf{r})$ ;
5    $\mathbf{S}_y = \mathbf{r}_{\text{sort}}\mathbf{H}^T$ ;
6    $le = 1$ ; // loop counter initialization
7   while ( $\text{sum}(\mathbf{S}_y) > 0$ ) & ( $le \leq n_{\Phi_e}$ ) do
8      $\mathbf{f}_{e_{pv}} = \Phi_e(le)$ ;
9      $\mathbf{r}_t = \mathbf{r}_{\text{sort}} \oplus \mathbf{f}_{e_{pv}}$ ;
10     $\mathbf{r}_o = \pi_1^{-1}(\mathbf{r}_t)$ ;
11     $\mathbf{S}_y = \mathbf{r}_o\mathbf{H}^T$ ;
12     $le = le + 1$ ;
13  end
14 else
15    $\mathbf{r}_o = \mathbf{r}$ 
16 end
17 if ( $\text{sum}(\mathbf{S}_y) == 0$ ) then
18    $\mathbf{c}_o^{\text{opt}} = \mathbf{r}_o$ ;
19 else
20    $\mathbf{c}_o^{\text{opt}} = \mathbf{r}$ ;
21 end

```

in the case of  $BCH(15, 7)$  are tabulated in Table XI. Figure 16 displays a scatter plot illustrating the coding gain of EDFD for various values of  $\epsilon$  at  $E_b/N_0$  of  $10^{-5}$ . Figure 17 illustrates the BER performance against the worst-case complexity of EDFD for various  $\epsilon$  values. EDFD provides varying BER performance levels for different complexities, depending on the chosen  $\epsilon$ .

## VIII. Conclusions and Discussion

The coding gains at BERs of  $10^{-5}$  and  $10^{-6}$ , as detailed in Table XII, highlight the performance of the proposed DFD and other decoders for the  $BCH(127, 113)$  code. Specifically, the proposed DFD exhibits a coding gain advantage of 0.6dB and 1.4dB over GRAND [53] at BERs of  $10^{-5}$  and  $10^{-6}$ , respectively, despite its significantly lower complexity.

In pursuit of improved BER performance, we introduce the Extended Diversity Flip Decoder (EDFD). Table XI delineates the coding gains facilitated by EDFD. Remarkably,



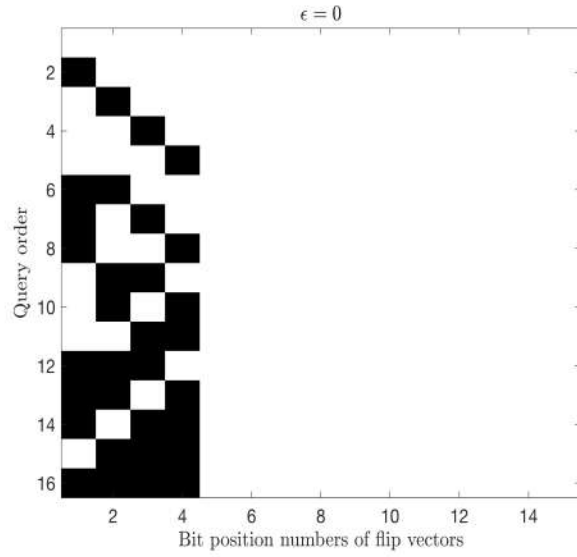


FIGURE 11: Flip vectors pattern of EDFD with  $\epsilon = 0$ : BCH(15, 7)

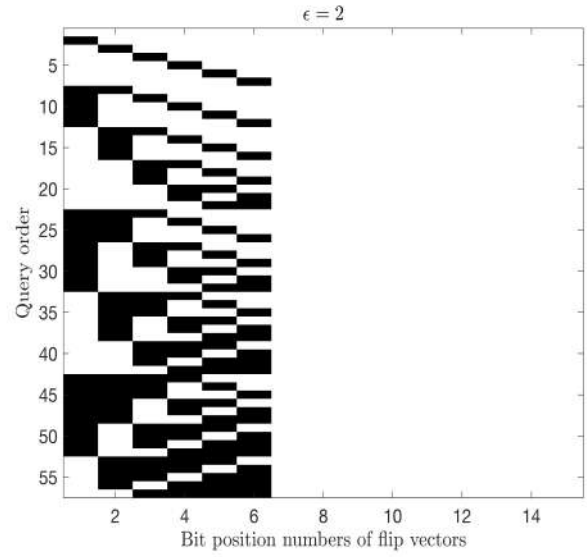


FIGURE 13: Flip vectors pattern of EDFD with  $\epsilon = 2$ : BCH(15, 7)

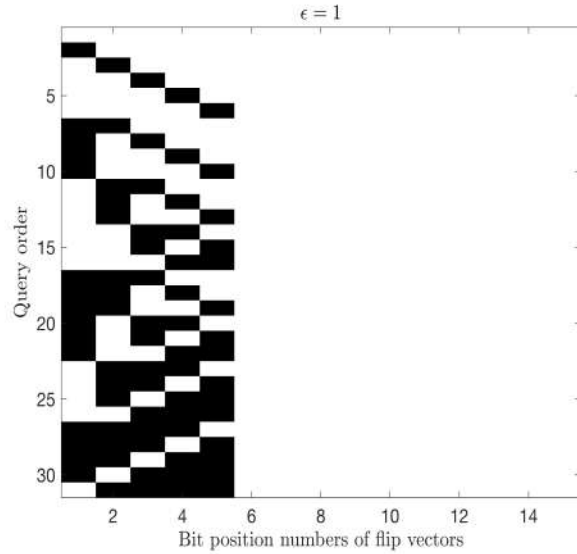


FIGURE 12: Flip vectors pattern of EDFD with  $\epsilon = 1$ : BCH(15, 7)

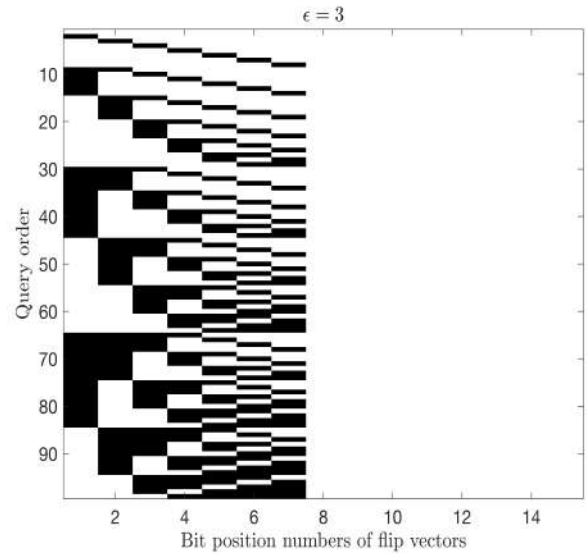


FIGURE 14: Flip vectors pattern of EDFD with  $\epsilon = 3$ : BCH(15, 7)

observe that as the  $\epsilon$  value increases, the BER performance provided by EDFD improves.

In conclusion, the proposed DFD exhibits several key attributes that make it a compelling decoding solution:

- **Universality:** Universally applicable for diverse linear codes.
- **Preserves diversity in fading channels:** The DFD is capable of preserving diversity order in challenging fading channel conditions.

- **Significantly lower complexity compared to existing approaches:** The proposed DFD imposes a significantly lower complexity than the existing methods making it ideal for low-complexity applications.

In our future research, we will harness the DFD in diverse dispersive high-Doppler system scenarios of both classical and quantum-coded scenarios.

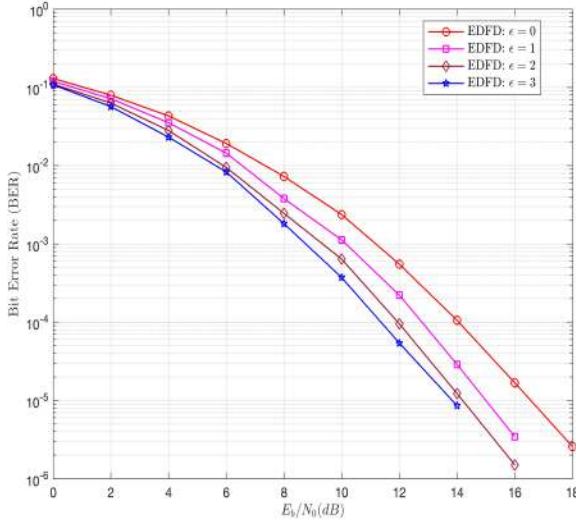


FIGURE 15: BER simulation using EDFD: BCH(15, 7)

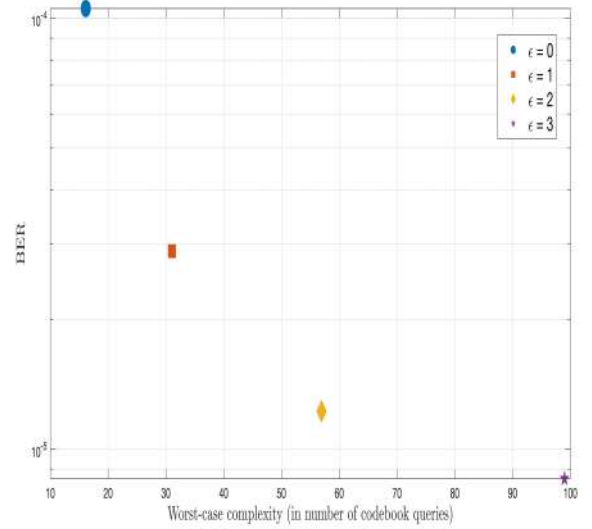


FIGURE 17: Scatter plot of BER (at  $E_b/N_0$  of 14dB) versus worst-case complexity (in number of codebook queries) of EDFD for different extension window lengths ( $\epsilon$ ): BCH(15, 7)

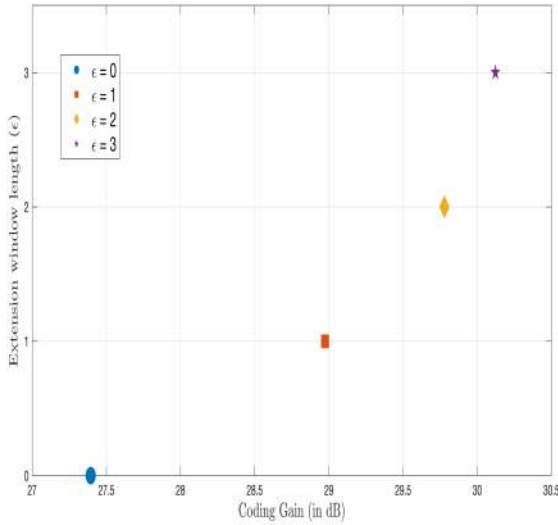


FIGURE 16: Scatter plot of coding gain of EDFD at BER of  $10^{-5}$  for different extension window lengths ( $\epsilon$ ): BCH(15, 7)

### Acknowledgment

The authors thank Prof. M. Medard and Prof. K. Duffy for their valuable discussions regarding this paper.

### REFERENCES

- [1] J. Huang, D. Fang, Y. Qian, and R. Q. Hu, "Recent Advances and Challenges in Security and Privacy for V2X Communications," *IEEE Open Journal of Vehicular Technology*, vol. 1, pp. 244–266, 2020.
- [2] A. Hajisami, J. Lansford, A. Dingankar, and J. Misener, "A Tutorial on the LTE-V2X Direct Communication," *IEEE Open Journal of Vehicular Technology*, vol. 3, pp. 388–398, 2022.
- [3] A. Molina-Galan, L. Lusvardi, B. Coll-Perales, J. Gozalvez, and M. L. Merani, "On the Impact of Re-evaluation in 5G NR V2X Mode

- 2," *IEEE Transactions on Vehicular Technology*, pp. 1–15, 2023.
- [4] S. Chen, J. Hu, Y. Shi, Y. Peng, J. Fang, R. Zhao, and L. Zhao, "Vehicle-to-Everything (V2X) Services Supported by LTE-Based Systems and 5G," *IEEE Comm. Standards Magazine*, vol. 1, no. 2, pp. 70–76, 2017.
- [5] Z. Li, T. Yu, R. Fukatsu, G. K. Tran, and K. Sakaguchi, "Towards Safe Automated Driving: Design of Software-Defined Dynamic MmWave V2X Networks and PoC Implementation," *IEEE Open Journal of Vehicular Technology*, vol. 2, pp. 78–93, 2021.
- [6] S.-Y. Lien and H.-L. Tsai, "3GPP V2X on Unlicensed Spectrum: Performance Analysis and Optimum Channel Access Strategies," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 9, pp. 9230–9243, 2021.
- [7] 3GPP TR 22.185, v.14.2.1, "Service Requirements for V2X Services," <https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=2989>, Nov. 2016.
- [8] 3GPP, "3rd Generation Partnership Project," <https://www.3gpp.org/>.
- [9] Z. Ma, M. Xiao, Y. Xiao, Z. Pang, H. V. Poor, and B. Vucetic, "High-Reliability and Low-Latency Wireless Communication for Internet of Things: Challenges, Fundamentals, and Enabling Technologies," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 7946–7970, 2019.

TABLE XII: Coding gain of different decoders over uncorrelated Rayleigh fading channel: BCH(127, 113)

Decoder	$E_b/N_0$ at BER of		Coding Gain at BER of	
	$10^{-5}$	$10^{-6}$	$10^{-5}$	$10^{-6}$
Uncoded	41.4dB	51.4dB	0dB	0dB
GRAND [53]	22dB	25.2dB	19.4dB	26.2dB
Fading GRAND [60]	16.2dB	18.4dB	25.2dB	33dB
ORBGRAND-psoft [61]	14.1dB	16dB	27.3dB	35.4dB
ORBGRAND [58]	13.2dB	15dB	28.2dB	36.4dB
Proposed DFD	21.4dB	23.8dB	20dB	27.6dB

- [10] O. A. Amodu and M. Othman, "Machine-to-Machine Communication: An Overview of Opportunities," *Computer Networks*, vol. 145, pp. 255–276, 2018.
- [11] C. Bockelmann, N. Pratas, H. Nikopour, K. Au, T. Svensson, C. Stefanovic, P. Popovski, and A. Dekorsy, "Massive machine-type communications in 5G: physical and MAC-layer solutions," *IEEE Communications Magazine*, vol. 54, no. 9, pp. 59–65, 2016.
- [12] G. Durisi, T. Koch, and P. Popovski, "Toward Massive, Ultrareliable, and Low-Latency Wireless Communication With Short Packets," *Proceedings of the IEEE*, vol. 104, no. 9, pp. 1711–1726, 2016.
- [13] P. Nouri, H. Alves, M. A. Uusitalo, O. Alcaraz López, and M. Latvaaho, "Machine-type Wireless Communications Enablers for Beyond 5G: Enabling URLLC via diversity under hard deadlines," *Computer Networks*, vol. 174, p. 107227, 2020.
- [14] F. A. Monteiro, O. L. A. López, and H. Alves, "Massive Wireless Energy Transfer With Statistical CSI Beamforming," *IEEE Journal of Selected Topics in Signal Processing*, vol. 15, no. 5, pp. 1169–1184, 2021.
- [15] M. Shirvanimoghaddam, M. S. Mohammadi, R. Abbas, A. Minja, C. Yue, B. Matuz, G. Han, Z. Lin, W. Liu, Y. Li, S. Johnson, and B. Vucetic, "Short Block-Length Codes for Ultra-Reliable Low Latency Communications," *IEEE Communications Magazine*, vol. 57, no. 2, pp. 130–137, 2019.
- [16] R. C. Bose and D. K. Ray-Chaudhuri, "On a Class of Error Correcting Binary Group Codes," *Information and control*, vol. 3, no. 1, pp. 68–79, 1960.
- [17] P. Pfeifer and H. T. Vierhaus, "Forward Error Correction in Wireless Communication Systems for Industrial Applications," in *2017 Signal Processing: Algorithms, Architectures, Arrangements, and Applications (SPA)*, 2017, pp. 14–14.
- [18] M. Bossert, R. Schulz, and S. Bitzer, "On Hard and Soft Decision Decoding of BCH Codes," *IEEE Transactions on Information Theory*, vol. 68, no. 11, pp. 7107–7124, 2022.
- [19] M. Baldi, F. Chiaraluce *et al.*, "A Simple Scheme for Belief Propagation Decoding of BCH and RS Codes in Multimedia Transmissions," *International Journal of Digital Multimedia Broadcasting*, vol. 2008, 2008.
- [20] G. Liva, L. Gaudio, T. Ninacs, and T. Jerkovits, "Code Design for Short Blocks: A Survey," *arXiv preprint arXiv:1610.00873*, 2016.
- [21] A. Becker, A. Joux, A. May, and A. Meurer, "Decoding Random Binary Linear Codes in  $2^{n/20}$ : How  $1+1=0$  Improves Information Set Decoding," in *Advances in Cryptology – EUROCRYPT 2012*, D. Pointcheval and T. Johansson, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 520–536.
- [22] I. Chatzigeorgiou and F. A. Monteiro, "Symbol-Level GRAND for High-Order Modulation Over Block Fading Channels," *IEEE Communications Letters*, vol. 27, no. 2, pp. 447–451, 2023.
- [23] L. Zheng and D. Tse, "Diversity and Multiplexing: A Fundamental Tradeoff in Multiple-Antenna Channels," *IEEE Transactions on Information Theory*, vol. 49, no. 5, pp. 1073–1096, 2003.
- [24] A. A. Patel, A. M. Ahmed, B. P. Sai, and M. Z. A. Khan, "Parity Check Codes for Second Order Diversity," *IETE Technical Review*, vol. 0, no. 0, pp. 1–9, 2023.
- [25] J. Wolf, "Efficient Maximum Likelihood Decoding of Linear Block Codes Using a Trellis," *IEEE Transactions on Information Theory*, vol. 24, no. 1, pp. 76–80, 1978.
- [26] H. Tanaka and K. Kakigahara, "Simplified Correlation Decoding by Selecting Possible Codewords Using Erasure Information (Corresp.)," *IEEE Transactions on Information Theory*, vol. 29, no. 5, pp. 743–748, 1983.
- [27] G. Forney, "Generalized Minimum Distance Decoding," *IEEE Transactions on Information Theory*, vol. 12, no. 2, pp. 125–131, 1966.
- [28] J. Snyders and Y. Be'ery, "Maximum Likelihood Soft Decoding of Binary Block Codes and Decoders for the Golay Codes," *IEEE Transactions on Information Theory*, vol. 35, no. 5, pp. 963–975, 1989.
- [29] N. Lous, P. Bours, and H. van Tilborg, "On Maximum Likelihood Soft-Decision Decoding of Binary Linear Codes," *IEEE Transactions on Information Theory*, vol. 39, no. 1, pp. 197–203, 1993.
- [30] J. Snyders, "Reduced Lists of Error Patterns for Maximum Likelihood Soft Decoding," *IEEE Transactions on Information Theory*, vol. 37, no. 4, pp. 1194–1200, 1991.
- [31] J. Snyders, "On Survivor Error Patterns For Maximum Likelihood Soft Decoding," in *Proceedings of the 1991 IEEE International Symposium on Information Theory*, 1991, pp. 192–192.
- [32] G. Forney, "Coset Codes. II. Binary Lattices and Related Codes," *IEEE Transactions on Information Theory*, vol. 34, no. 5, pp. 1152–1187, 1988.
- [33] D. Muder, "Minimal Trellises for Block Codes," *IEEE Transactions on Information Theory*, vol. 34, no. 5, pp. 1049–1053, 1988.
- [34] T. Kasami, T. Takata, T. Fujiwara, and S. Lin, "On Complexity of Trellis Structure of Linear Block Codes," *IEEE Transactions on Information Theory*, vol. 39, no. 3, pp. 1057–1064, 1993.
- [35] M. F. S. Lin, "Soft-Decision Decoding of Linear Block Codes Based on Ordered Statistics," *IEEE Transactions on Information Theory*, vol. 41, no. 5, pp. 1379–1396, 1995.
- [36] Y. Wu and C. N. Hadjicostis, "Soft-Decision Decoding of Linear Block Codes Using Preprocessing and Diversification," *IEEE Transactions on Information Theory*, vol. 53, no. 1, pp. 378–393, 2007.
- [37] W. Jin and M. Fossorier, "Probabilistic Sufficient Conditions on Optimality for Reliability Based Decoding of Linear Block Codes," in *2006 IEEE International Symposium on Information Theory*, 2006, pp. 2235–2239.
- [38] H. Tang, Y. Liu, M. Fossorier, and S. Lin, "On Combining Chase-2 and GMD Decoding Algorithms for Nonbinary Block Codes," *IEEE Communications Letters*, vol. 5, no. 5, pp. 209–211, 2001.
- [39] R. Koetter and A. Vardy, "Algebraic Soft-Decision Decoding of Reed-Solomon Codes," *IEEE Transactions on Information Theory*, vol. 49, no. 11, pp. 2809–2825, 2003.
- [40] J. Jiang and K. R. Narayanan, "Algebraic Soft-Decision Decoding of Reed-Solomon Codes Using Bit-Level Soft Information," *IEEE transactions on information theory*, vol. 54, no. 9, pp. 3907–3928, 2008.
- [41] A. Vardy and Y. Be'ery, "Bit-Level Soft-Decision Decoding of Reed-Solomon Codes," *IEEE Transactions on Communications*, vol. 39, no. 3, pp. 440–444, 1991.
- [42] W. Jin and M. P. Fossorier, "Iterative Biased Reliability-Based Decoding of Binary Linear Codes," = *IEICE technical report* : , vol. 105, no. 85, pp. 65–70, 2005.
- [43] I. Dumer, "Information-Set Soft-Decision Decoding," in *1998 Information Theory Workshop (Cat. No.98EX131)*, 1998, pp. 77–78.
- [44] S. Bitzer and M. Bossert, "On Multibasis Information Set Decoding," in *2022 IEEE International Symposium on Information Theory (ISIT)*, 2022, pp. 2780–2784.
- [45] M. Bossert and F. Hergert, "Hard- and Soft-Decision Decoding Beyond the Half Minimum Distance—An Algorithm for Linear Codes (Corresp.)," *IEEE Transactions on Information Theory*, vol. 32, no. 5, pp. 709–714, 1986.
- [46] D. Wu, Y. Li, X. Guo, and Y. Sun, "Ordered Statistic Decoding for Short Polar Codes," *IEEE Communications Letters*, vol. 20, no. 6, pp. 1064–1067, 2016.
- [47] A. Kothiyal, O. Y. Takeshita, W. Jin, and M. Fossorier, "Iterative Reliability-Based Decoding of Linear Block Codes with Adaptive Belief Propagation," *IEEE Communications Letters*, vol. 9, no. 12, pp. 1067–1069, 2005.

- [48] M. Bossert, "An Iterative Hard and Soft Decision Decoding Algorithm for Cyclic Codes," in *SCC 2019; 12th International ITG Conference on Systems, Communications and Coding*. VDE, 2019, pp. 1–6.
- [49] W. Jin and M. P. Fossorier, "Reliability-Based Soft-Decision Decoding with Multiple Biases," *IEEE transactions on information theory*, vol. 53, no. 1, pp. 105–120, 2006.
- [50] M. Bossert, R. Schulz, and S. Bitzer, "On Information Set Decoding of BCH Codes over Binary Symmetric Channel," in *Proc. IEEE Int. Workshop Math. Coding*, 2020.
- [51] J. Conway and N. Sloane, "Soft Decoding Techniques for Codes and Lattices, Including the Golay Code and the Leech Lattice," *IEEE Transactions on Information Theory*, vol. 32, no. 1, pp. 41–50, 1986.
- [52] D. Chase, "Class of Algorithms for Decoding Block Codes with Channel Measurement Information," *IEEE Transactions on Information Theory*, vol. 18, no. 1, pp. 170–182, 1972.
- [53] K. R. Duffy, J. Li, and M. Médard, "Guessing Noise, Not Code Words," in *2018 IEEE International Symposium on Information Theory (ISIT)*, 2018, pp. 671–675.
- [54] K. R. Duffy, J. Li, and M. Medard, "Capacity-Achieving Guessing Random Additive Noise Decoding," *IEEE Transactions on Information Theory*, vol. 65, pp. 4023–4040, 7 2019.
- [55] K. R. Duffy and M. Médard, "Guessing Random Additive Noise Decoding with Soft Detection Symbol Reliability Information - SGRAND," in *2019 IEEE International Symposium on Information Theory (ISIT)*, 2019, pp. 480–484.
- [56] K. R. Duffy, A. Solomon, K. M. Konwar, and M. Médard, "5G NR CA-Polar Maximum Likelihood Decoding by GRAND," in *2020 54th Annual Conference on Information Sciences and Systems (CISS)*, 2020, pp. 1–5.
- [57] K. R. Duffy, "Ordered Reliability Bits Guessing Random Additive Noise Decoding," in *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 8268–8272.
- [58] K. R. Duffy, W. An, and M. Médard, "Ordered Reliability Bits Guessing Random Additive Noise Decoding," *IEEE Transactions on Signal Processing*, vol. 70, pp. 4528–4542, 2022.
- [59] A. Solomon, K. R. Duffy, and M. Médard, "Soft Maximum Likelihood Decoding using GRAND," in *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*, 2020, pp. 1–6.
- [60] S. M. Abbas, M. Jalaleddine, and W. J. Gross, "GRAND for Rayleigh Fading Channels," in *2022 IEEE Globecom Workshops (GC Wkshps)*, 2022, pp. 504–509.
- [61] H. Saeeddeen, M. Médard, and K. R. Duffy, "GRAND for Fading Channels using Pseudo-soft Information," in *GLOBECOM 2022 - 2022 IEEE Global Communications Conference*, 2022, pp. 3502–3507.
- [62] I. Chatzigeorgiou and F. A. Monteiro, "Symbol-Level GRAND for High-Order Modulation Over Block Fading Channels," *IEEE Communications Letters*, vol. 27, no. 2, pp. 447–451, 2023.
- [63] S. Allahkaram, F. A. Monteiro, and I. Chatzigeorgiou, "Symbol-Level Noise-Guessing Decoding with Antenna Sorting for URLLC Massive MIMO," 2023.
- [64] Allahkaram, Sahar and Monteiro, Francisco A. and Chatzigeorgiou, Ioannis, "URLLC with Coded Massive MIMO via Random Linear Codes and GRAND," in *2022 IEEE 96th Vehicular Technology Conference (VTC2022-Fall)*, 2022, pp. 1–5.
- [65] W. Liang and H. Liu, "Low-Complexity Error Correction Algorithm for Cyclic Redundancy Check Codes," in *2021 7th International Conference on Computer and Communications (ICCC)*, 2021, pp. 22–26.
- [66] A. A. Patel, A. M. Ahmed, and M. Z. A. Khan, "Parity Check Codes for Second Order Diversity," 2020, arXiv:2012.09497 [cs.IT].
- [67] L. Shu, *Error Control Coding*. Pearson India, 2011. [Online]. Available: [https://books.google.co.in/books?id=BG08Zylo\\_7gC](https://books.google.co.in/books?id=BG08Zylo_7gC)
- [68] A. M. Ahmed, "Multi-Radio Network Diversity for Next Generation Networks," Ph.D. dissertation, Indian Institute of Technology, Hyderabad, 2021.
- [69] H.-J. Kwon, J. Jeon, A. Bhorkar, Q. Ye, H. Harada, Y. Jiang, L. Liu, S. Nagata, B. L. Ng, T. Novlan, J. Oh, and W. Yi, "Licensed-Assisted Access to Unlicensed Spectrum in LTE Release 13," *IEEE Communications Magazine*, vol. 55, no. 2, pp. 201–207, 2017.
- [70] M. Labib, V. Marojevic, J. H. Reed, and A. I. Zaghoul, "Extending LTE into the Unlicensed Spectrum: Technical Analysis of the Proposed Variants," *IEEE Communications Standards Magazine*, vol. 1, no. 4, pp. 31–39, 2017.
- [71] Rojeena Bajracharya, Rakesh Shrestha, Yousaf Bin Zikria and Sung Won Kim, "LTE in the Unlicensed Spectrum: A Survey," *IETE Technical Review*, vol. 35, no. 1, pp. 78–90, 2018. [Online]. Available: <https://doi.org/10.1080/02564602.2016.1251344>
- [72] D. Kakadia, J. Yang, and A. Gilgur, *Evolved Universal Terrestrial Radio Access Network (EUTRAN)*, 09 2017, pp. 61–81.
- [73] A. Garcia-Saavedra, P. Patras, V. Valls, X. Costa-Perez, and D. J. Leith, "ORLA/OLAA: Orthogonal Coexistence of LAA and WiFi in Unlicensed Spectrum," *IEEE/ACM Transactions on Networking*, vol. 26, no. 6, pp. 2665–2678, 2018.
- [74] C. Cano, D. J. Leith, A. Garcia-Saavedra, and P. Serrano, "Fair Coexistence of Scheduled and Random Access Wireless Networks: Unlicensed LTE/WiFi," *IEEE/ACM Transactions on Networking*, vol. 25, no. 6, pp. 3267–3281, 2017.
- [75] X. Sun and L. Dai, "Towards Fair and Efficient Spectrum Sharing Between LTE and WiFi in Unlicensed Bands: Fairness-Constrained Throughput Maximization," *IEEE Transactions on Wireless Communications*, vol. 19, no. 4, pp. 2713–2727, 2020.
- [76] Y. Gao and S. Roy, "Achieving Proportional Fairness for LTE-LAA and Wi-Fi Coexistence in Unlicensed Spectrum," *IEEE Transactions on Wireless Communications*, vol. 19, no. 5, pp. 3390–3404, 2020.
- [77] H. Cui, V. C. M. Leung, S. Li, and X. Wang, "LTE in the Unlicensed Band: Overview, Challenges, and Opportunities," *IEEE Wireless Communications*, vol. 24, no. 4, pp. 99–105, 2017.
- [78] LIU, RUI and CHEN, QIMEI and YU, GUANDING and LI, GE-OFFREY YE and DING, ZHI, "Resource Management in LTE-U Systems: Past, Present, and Future," *IEEE Open Journal of Vehicular Technology*, vol. 1, pp. 1–17, 2020.
- [79] G. Naik, J.-M. Park, J. Ashdown, and W. Lehr, "Next Generation Wi-Fi and 5G NR-U in the 6 GHz Bands: Opportunities and Challenges," *IEEE Access*, vol. 8, pp. 153 027–153 056, 2020.
- [80] N. Patriciello, S. Lagén, B. Bojović, and L. Giupponi, "NR-U and IEEE 802.11 Technologies Coexistence in Unlicensed mmWave Spectrum: Models and Evaluation," *IEEE Access*, vol. 8, pp. 71 254–71 271, 2020.
- [81] Y. Jiang, J. Guo, and Z. Fei, "Performance Analysis of the Coexistence of 5G NR-Unlicensed and Wi-Fi with Mode Selection," in *2020 IEEE/CIC International Conference on Communications in China (ICCC)*, 2020, pp. 953–958.
- [82] A. Aijaz, "Packet Duplication in Dual Connectivity Enabled 5G Wireless Networks: Overview and Challenges," *IEEE Communications Standards Magazine*, vol. 3, no. 3, pp. 20–28, 2019.
- [83] A. M. Ahmed, A. Patel, and M. Z. A. Khan, "Parity Check Coded Super-MAC for Reliability Enhancements in Next-Generation Networks," in *2024 16th International Conference on COMMunication Systems NETWORKS (COMSNETS)*, 2024, pp. 1116–1121.
- [84] Bere Praveen Sai and Mohammed Zafar Ali Khan, "Bit Switching Decoding of Cyclic Hamming Codes for IoT Applications," in *2022 IEEE World AI IoT Congress (AIoT)*, 2022, pp. 743–748.
- [85] B. Praveen Sai and Mohammed Zafar Ali Khan, "Low Complexity, Diversity-Preserving Hard Decision Decoder for CRC Codes with IoT Applications," in *2022 IEEE 96th Vehicular Technology Conference (VTC2022-Fall)*, 2022, pp. 1–5.
- [86] Y. Wu and C. N. Hadjicostis, "Soft-Decision Decoding of Linear Block Codes Using Preprocessing and Diversification," *IEEE Transactions on Information Theory*, vol. 53, no. 1, pp. 378–393, 2007.
- [87] P. Massoud Salehi and J. Proakis, *Digital Communications*. McGraw-Hill Education, 2007.
- [88] C. Rose and M. Smith, *Mathematical Statistics with Mathematica: Mathematical Statistics with Mathematica*, ser. Mathematical Statistics with Mathematica. Springer, 2002.
- [89] H.-C. Lee, "An Efficient BCH Decoder for WBAN Applications," *IEEE Communications Letters*, vol. 25, no. 6, pp. 1766–1770, 2021.
- [90] T. K. Moon, *Error Correction Coding: Mathematical Methods and Algorithms*, 1st ed. John Wiley & Sons, 2005.