

# Low-complexity Block-Based Decoding Algorithms for Short Block Channels

Mody Sy<sup>1</sup> and Raymond Knopp<sup>1</sup>

EURECOM, 06410 BIOT, France

mody.sy@eurecom.fr, raymond.knopp@eurecom.fr

**Abstract**—This paper presents low-complexity block-based encoding and decoding algorithms for short block length channels. In terms of the precise use-case, we are primarily concerned with the baseline 3GPP Short block transmissions in which payloads are encoded by Reed-Muller codes and paired with orthogonal DMRS. In contemporary communication systems, the short block decoding often employs the utilization of DMRS-based least squares channel estimation, followed by maximum likelihood decoding. However, this methodology can incur substantial computational complexity when processing long bit length codes. We propose an innovative approach to tackle this challenge by introducing the principle of block/segment encoding using First-Order RM Codes which is amenable to low-cost decoding through block-based fast Hadamard transforms. The Block-based FHT has demonstrated to be cost-efficient with regards to decoding time, as it evolves from quadric to quasi-linear complexity with a manageable decline in performance. Additionally, by incorporating an adaptive DMRS/data power adjustment technique, we can bridge/reduce the performance gap and attain high sensitivity, leading to a good trade-off between performance and complexity to efficiently handle small payloads.

**Index Terms**—5G NR, Short block-lengths, ML detection, Training-based Transmission, Reed Muller codes, Fast Hadamard Transform, Block-based Encoding and Decoding, Adaptive Power Adjustment.

## I. INTRODUCTION

5G NR aims to effectively transmit small payloads, typically consisting of tens of bits, with low error rates even in low signal-to-noise environments. This requires the design of strong structured coding strategies and low-complexity decoding algorithms to meet the demanding requirements of URLLC use cases. In this paper, we explore potential methods for improving the detection and decoding performance with low computational complexity in 5G/6G systems. Our focus is on the use-case of 3GPP short block transmissions paired with orthogonal *Demodulation Reference Signals* or DMRS, which employ Reed-Muller coding schemes. Thus, the DMRS symbols, which are message-independent, are utilized by the gNodeB receiver for resolving channel uncertainty through explicit channel estimation or joint estimation and detection. In 3GPP systems, short blocks, ranging from 3 to 11 bits, are encoded using a family of Reed-Muller codes of dimension  $\mathcal{C}(32, K)$  prior to being transmitted on the uplink control channels [1]. On reception, the messages are recovered using maximum likelihood decoding, which can be computationally demanding.

Reed-Muller codes (RM codes) are commonly known to be decodable using *Hadamard Transform* (HT) or *Fast Hadamard Transform* (FHT). However, it is well-established that decoding the first-order RM code ( $RM(r = 1, m)$ ) using FHT is easier compared to higher-order RM codes ( $r \geq 2$ ). In recent literature, several innovative algorithms have been

proposed for decoding RM codes of any order [2]–[6].

Although maximum likelihood decoding algorithms have been extensively investigated in traditional literature for decoding data packets encoded with first-order Reed-Muller codes [7], it can become computationally expensive when the message length exceeds 6 bits. This is because the resulting codewords tend to be excessively long, leading to complex decoding processes that involve high-dimensional Hadamard transforms when using a FHT-based decoder. This presents a significant challenge for transmitting short packets as the cost can be substantial. As an illustration, for a message of  $K = 11$  bits, the length of the code words would be  $N' = 2^{10}$  bits using a first order  $RM(1, m = 10)$ . Hence, to address the constraint of having a message length of  $K \geq 6$  bits, we can utilize the principle of encoding and decoding by blocks. This method takes advantage of the low complexity decoding offered by FHT-based decoders.

The objective is to segment the message into smaller, more manageable segments of bits, which can then be fed into  $RM(1, m)$  encoders and concatenated. Upon reception, the received code is deconcatenated and decoded through the appropriate dimension of the Hadamard Transform which is amenable to a low complexity receiver.

The manuscript is structured as follows. Section II lays out the system model foundation, Section III highlights the proposed encoding/decoding methods, Section IV presents the results and performance analysis, and finally Section V concludes the paper.

## II. SYSTEM MODEL

Consider a discrete-time model in which the transmitted and received symbols are  $N$ -dimensional column vectors, and thus a system is designed in such a way that the relationship between the transmitted and received signals is as follows:

$$\mathbf{y} = \mathbf{h}\mathbf{x} + \mathbf{z}, \quad (1)$$

where  $\mathbf{y}$  represents an observed vector in  $N$  complex dimensions,  $\mathbf{x}$  is a  $N$ -dimensional modulated vector,  $\mathbf{z}$  is additive white Gaussian noise whose the real and imaginary components are independent and have variance  $\sigma^2$ . The transmitted vector is composed of data-independent components, known as pilot or reference signals, that help resolve channel ambiguity in time, frequency, and space. These reference signals are used to estimate the vector channels  $\mathbf{h}$  and are *interleaved* with data-dependent components in accordance with the attributes of the propagation channel.

$$\mathbf{x} = \mathbf{x}^{(d)} + \mathbf{x}^{(p)} \quad (2)$$

where subscripts  $(d)$  and  $(p)$  serve to denote the data components and reference signals respectively. The number of data

dimensions is denoted by  $N_d$ , and the number of reference signal dimensions is denoted by  $N_p$ , where  $N_d + N_p = N$ . In 3GPP, the standard notation for  $N$  is  $12\mathcal{P}\mathcal{L}$ , where  $\mathcal{P}$  refers to the number of physical resource blocks, each consisting of 12 complex dimensions or resource elements. The value of  $\mathcal{P}$  usually falls within a range of 1 to 16.  $\mathcal{L}$  represents the number of symbols, typically ranging from 1 to 14, but it can be increased if multiple slots are utilized to signal the channel bits.

### III. BLOCK-BASED CHANNEL CODING FOR SHORT DATA

#### A. Block-based Encoding Principle

In instances where the payload exceeds 6 bits, such as in the case of  $K = 11$  bits, a combination of two first-order Reed-Muller codes,  $RM(1, m = 4)$  and  $RM(1, m = 5)$ , can be employed to encode the respective sub-blocks of 5 bits and 6 bits.

In regards to the Reed-Muller code  $RM(1, 4)$ , the codewords are generated using (3).

$$\mathbf{c}^{(1)} = \mathbf{m}^{(1)}\mathbf{G}^{(1)} = \mathbf{m}^{(1)} \begin{bmatrix} \mathbf{1} & \mathbf{v}_4 & \mathbf{v}_3 & \mathbf{v}_2 & \mathbf{v}_1 \end{bmatrix}^T, \quad (3)$$

This code is characterized as an  $(N' = 16, K = 5, d_{min} = 8)$  code, where the minimum distance of the  $RM(r, m)$  is defined as  $2^{m-r}$ . The monomials of degree less than or equal to  $r$  are represented by  $\mathbf{1}, \mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3, \mathbf{v}_4$ , with associated vectors.

| $\mathbf{1}$ | $\mathbf{v}_4$ | $\mathbf{v}_3$ | $\mathbf{v}_2$ | $\mathbf{v}_1$ |
|--------------|----------------|----------------|----------------|----------------|
| 1            | 0              | 0              | 0              | 0              |
| 1            | 0              | 0              | 0              | 1              |
| 1            | 0              | 0              | 1              | 0              |
| 1            | 0              | 0              | 1              | 1              |
| 1            | 0              | 1              | 0              | 0              |
| 1            | 0              | 1              | 0              | 1              |
| 1            | 0              | 1              | 1              | 0              |
| 1            | 0              | 1              | 1              | 1              |
| 1            | 1              | 0              | 0              | 0              |
| 1            | 1              | 0              | 0              | 1              |
| 1            | 1              | 0              | 1              | 0              |
| 1            | 1              | 0              | 1              | 1              |
| 1            | 1              | 1              | 0              | 0              |
| 1            | 1              | 1              | 0              | 1              |
| 1            | 1              | 1              | 1              | 0              |
| 1            | 1              | 1              | 1              | 1              |

If we consider  $RM(1, 5)$ , code words are generated by

$$\mathbf{c}^{(2)} = \mathbf{m}^{(2)}\mathbf{G}^{(2)} = \mathbf{m}^{(2)} \begin{bmatrix} \mathbf{1} & \mathbf{v}_5 & \mathbf{v}_4 & \mathbf{v}_3 & \mathbf{v}_2 & \mathbf{v}_1 \end{bmatrix}^T.$$

This is a  $(N' = 32, K = 6, d_{min} = 16)$  code.

Therefore, the process of concatenation entails the merging of the two sub-codes

$$\mathbf{c} = [\mathbf{c}^{(1)} \ \mathbf{c}^{(2)}] = [\mathbf{m}^{(1)}\mathbf{G}^{(1)} \ \mathbf{m}^{(2)}\mathbf{G}^{(2)}]. \quad (4)$$

The resultant bit sequence, prior to the concatenation of code blocks, is represented as  $c(0), c(1), c(2), \dots, c(N' - 1)$ . The output bit sequence, following rate matching, is denoted as  $e(0), e(1), e(2), \dots, e(E-1)$ , where the length of the rate matching output sequence,  $E$ , is dependent on the number of physical resource blocks or (PRBs). This information block, represented as  $\mathbf{e}$ , is then subjected to quadrature phase shift keying (QPSK) modulation, resulting in a block of complex-valued modulation symbols  $x(0), x(1), \dots, x(E/2 - 1)$ . The resource mapping process follows, whose aim is to allocate the modulated symbols onto resource occasions.

The procedure of block-based RM encoding is depicted in Figure 1.

#### B. Block-based Decoding via FHT

A Hadamard matrix, represented as  $\mathbf{H}_n$ , is a square matrix with dimensions  $n \times n$  consisting of elements of  $\pm 1$ . It satisfies the property that the matrix product of  $\mathbf{H}_n$  and its transpose,  $\mathbf{H}_n^T$ , results in a scalar multiple of the identity matrix,  $\mathbf{I}_n$ , where  $\mathbf{I}_n$  is an  $n \times n$  identity matrix and  $n$  is the order of the Hadamard matrix.

$$\mathbf{H}_n = \begin{bmatrix} \mathbf{H}_{n-1} & \mathbf{H}_{n-1} \\ \mathbf{H}_{n-1} & -\mathbf{H}_{n-1} \end{bmatrix}, \quad n = 1, 2, 4, 16, 32, \dots$$

Consider the received sequence  $\mathbf{u} = (u_0, u_1, \dots, u_{2^m-1}) \in \mathbb{F}_2$ , and let  $\mathbf{c} = (c_0, c_1, \dots, c_{2^m-1}) \in \mathbb{F}_2$  be a codeword. The bipolar representation of  $\mathbf{u}$  is denoted as  $\mathbf{U} \in \{-1, +1\}$  and is defined as  $\mathbf{U} = (-1)^{\mathbf{u}}$ . Similarly, the bipolar representation of  $\mathbf{c}$  is denoted as  $\mathbf{C}$  and defined as  $\mathbf{C} = (-1)^{\mathbf{c}}$ . The decoding algorithm involves computing the correlation between  $\mathbf{U}$  and  $\mathbf{C}_i$ , denoted as  $\Delta_i$ , for each of the  $2^{m-1}$  codewords  $\mathbf{C}_i = (-1)^{\mathbf{c}_i}$ . The final step is to select the codeword for which  $\Delta_i$  is the maximum. The simultaneous computation of all correlations can be depicted as a matrix representation. Denoting the column vector  $\mathbf{C}_i$  and constructing the matrix  $\mathbf{H} = [\mathbf{C}_0 \ \mathbf{C}_1 \ \dots \ \mathbf{C}_{2^m-1}]$ , the computation of all correlations can be expressed as follows:

$$\Delta = \mathbf{U}\mathbf{H}, \quad (5)$$

Where  $\mathbf{H}$  is a matrix of dimension  $2^m$ . For the first sub-block, which utilizes a  $RM(1, 4)$  code with generator matrix  $\mathbf{G}^{(1)}$ ,  $\mathbf{H}_{16}^{(1)}$  is employed. Similarly, for the second subblock, which employs a  $RM(1, 5)$  code with generator matrix  $\mathbf{G}^{(2)}$ ,  $\mathbf{H}_{32}^{(2)}$  is utilized. Furthermore, detailed expositions concerning the algorithms employed in the decoding process of first-order RM codes through the utilization of the Hadamard transform can be readily found within the scholarly works authored by Moon [8] and Wicker [9]. This decoding process can be optimized through the utilization of a FHT which is applicable to Hadamard matrices produced through the Sylvester construction. This optimization is based on the fact that  $\mathbf{H}_{2^m} = \mathbf{H}_2 \otimes \mathbf{H}_{2^{m-1}}$ , where the Kronecker product of matrices, denoted by  $\otimes$ , is applied. As a result, the matrix  $\mathbf{H}_{2^m}$  can be decomposed as stated in the theorem derived from linear algebraic principles [8].

$$\mathbf{H}_{2^m} = \mathbf{W}_{2^m}^{(1)}\mathbf{W}_{2^m}^{(2)} \dots \mathbf{W}_{2^m}^{(m)}, \quad (6)$$

where  $\mathbf{W}_{2^m}^{(i)} = \mathbf{I}_{2^{m-i}} \otimes \mathbf{H}_2 \otimes \mathbf{I}_{2^{i-1}}$ ,  $\mathbf{I}$  is an identity matrix. Thus it comes,

$$\begin{aligned} \mathbf{H}_{16}^{(1)} &= \mathbf{W}_{16}^{(1)}\mathbf{W}_{16}^{(2)}\mathbf{W}_{16}^{(3)}\mathbf{W}_{16}^{(4)} \\ &= (\mathbf{I}_{2^3} \otimes \mathbf{H}_2 \otimes \mathbf{I}_{2^0}) (\mathbf{I}_{2^2} \otimes \mathbf{H}_2 \otimes \mathbf{I}_{2^1}) \\ &\quad (\mathbf{I}_{2^1} \otimes \mathbf{H}_2 \otimes \mathbf{I}_{2^2}) (\mathbf{I}_{2^0} \otimes \mathbf{H}_2 \otimes \mathbf{I}_{2^3}). \end{aligned} \quad (7)$$

$$\begin{aligned} \mathbf{H}_{32}^{(2)} &= \mathbf{W}_{32}^{(1)}\mathbf{W}_{32}^{(2)}\mathbf{W}_{32}^{(3)}\mathbf{W}_{32}^{(4)}\mathbf{W}_{32}^{(5)} \\ &= (\mathbf{I}_{2^4} \otimes \mathbf{H}_2 \otimes \mathbf{I}_{2^0}) (\mathbf{I}_{2^3} \otimes \mathbf{H}_2 \otimes \mathbf{I}_{2^1}) (\mathbf{I}_{2^2} \otimes \mathbf{H}_2 \otimes \mathbf{I}_{2^2}) \\ &\quad (\mathbf{I}_{2^1} \otimes \mathbf{H}_2 \otimes \mathbf{I}_{2^3}) (\mathbf{I}_{2^0} \otimes \mathbf{H}_2 \otimes \mathbf{I}_{2^4}). \end{aligned} \quad (8)$$

Let's consider  $\mathbf{U}^{(1)} = [U_0, U_1, \dots, U_{15}]$  and  $\mathbf{U}^{(2)} = [U_0, U_1, \dots, U_{31}]$ , the received sequences to be fed to the

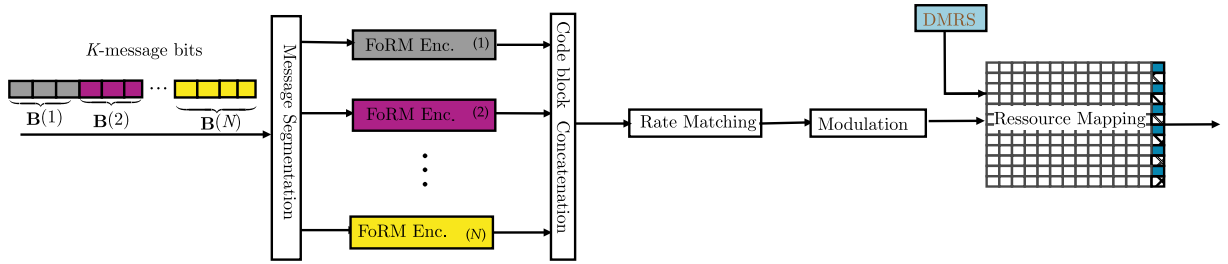


Fig. 1. Block-based RM(1, M) encoding of Short block -length :Transmitter end.

decoders. The corresponding Hadamard transforms can then be written as

$$\begin{aligned} \Delta^{(1)} &= \mathbf{U}^{(1)} \mathbf{H}_{16}^{(1)} = \mathbf{U}^{(1)} \left( \mathbf{W}_{16}^{(1)} \mathbf{W}_{16}^{(2)} \mathbf{W}_{16}^{(3)} \mathbf{W}_{16}^{(4)} \right), \\ \Delta^{(2)} &= \mathbf{U}^{(2)} \mathbf{H}_{32}^{(2)} = \mathbf{U}^{(2)} \left( \mathbf{W}_{32}^{(1)} \mathbf{W}_{32}^{(2)} \mathbf{W}_{32}^{(3)} \mathbf{W}_{32}^{(4)} \mathbf{W}_{32}^{(5)} \right). \end{aligned} \quad (9)$$

where

$$\begin{aligned} \mathbf{W}_{16}^{(1)} &= \mathbf{I}_8 \otimes \mathbf{H}_2, & \mathbf{W}_{32}^{(1)} &= \mathbf{I}_{16} \otimes \mathbf{H}_2, \\ \mathbf{W}_{16}^{(2)} &= \mathbf{I}_4 \otimes \mathbf{H}_2 \otimes \mathbf{I}_2, & \mathbf{W}_{32}^{(2)} &= \mathbf{I}_8 \otimes \mathbf{H}_2 \otimes \mathbf{I}_2 \\ \mathbf{W}_{16}^{(3)} &= \mathbf{I}_2 \otimes \mathbf{H}_2 \otimes \mathbf{I}_4, & \mathbf{W}_{32}^{(3)} &= \mathbf{I}_4 \otimes \mathbf{H}_2 \otimes \mathbf{I}_4, \\ \mathbf{W}_{16}^{(4)} &= \mathbf{H}_2 \otimes \mathbf{I}_8, & \mathbf{W}_{32}^{(4)} &= \mathbf{I}_2 \otimes \mathbf{H}_2 \otimes \mathbf{I}_8 \\ & & \mathbf{W}_{32}^{(5)} &= \mathbf{H}_2 \otimes \mathbf{I}_{16}. \end{aligned}$$

The conventional computation of the Hadamard transform  $\mathbf{H}_{2^m}$  results in  $2^m$  elements, each of which is obtained through  $2^m$  addition/subtraction operations. This leads to a computational complexity of  $(2^m)^2 = \mathcal{O}(N'^2)$ , which is equivalent to the complexity of a standard ML decoder that operates in a *quadratic order*. In contrast, the Fast Hadamard transform, which has  $m$  stages, has a computational complexity of  $m2^m = \mathcal{O}(N' \log N')$  (i.e., *quasi-linear complexity*) due to its  $2^m$  addition/subtraction operations per stage. The procedure of block-based RM decoding is therefore depicted in Figure 2.

#### IV. NUMERICAL RESULTS

For illustration purposes, we focus on PUCCH2-based short block lengths, which consist of 11 bits. PUCCH2 is configurable in terms of the resource usage, but we consider the simplest comprising of 2 groups of 12 dimensions or *resource elements*, so-called PRBs, making 24 dimensions which consist of 16 for data components, and 8 for the so-called DMRS, which are known symbols used for channel estimation and tracking. The following simulation results were performed under the assumption of Rayleigh flat fading channel, utilizing 2, 4 and 8 antenna configurations. The antenna ports were subjected to independent and identically distributed realizations, with no correlation modeling applied. Figure 3, demonstrate the merits of the proposed block-based Reed-Muller (RM) decoding approach employing the HT/FHT. These results are contrasted with the conventional RM decoding technique recommended by 3GPP, employing a *Maximum Likelihood* (ML) decoder, within the Rayleigh fading channel scenario.

The results presented suggest that the ML decoder exhibits improved performance, however, this advantage is accompanied by a significant increase in computational complexity,

even when utilizing the block decoding principle. In practical implementation, each received code block ( $\mathcal{C}(11, 48) = [\mathcal{C}_1(5, 16), \mathcal{C}_2(6, 32)]$ ) is processed through a corresponding decoder, followed by concatenation at the decoder output stage. It is important to acknowledge that the efficacy of block decoding through both HT and FHT is comparable, exhibiting a marginal deviation of 1.3 dB from the ML receiver at a BLER threshold of 1% when  $n_{R\text{x}}=4$ . Moreover, some of the performance gap with the ML receiver can be bridged or at least reduced.

In instances where we transmit reference and data symbols jointly in common OFDM symbols, as is the case for PUCCH or PUSCH or in some downlink control channels, there is the possibility to optimize the data-reference power ratio. This is possible for both downlink and uplink transmission without incurring a penalty in terms of peak power increase. Therefore, the performance disparity between the ML receiver and the FHT-based receiver can be mitigated/reduced by reconceiving the system as follows  $\mathbf{y} = (\mathbf{x}^{(d)} + \beta \mathbf{x}^{(p)}) \cdot \mathbf{h} + \mathbf{z}$ . This adaptive power adjustment procedure is contingent on the values of  $\beta$ . The DMRS power is to be slightly increased in a judicious manner since  $\beta$  must be perfectly calibrated to ensure compliance with potential radio frequency constraints. Figure 3 illustrates the observable performance enhancement achieved by varying  $\beta$  values, specifically  $\beta = \{1.25, 1.5, 1.75\}$ . The most significant performance gain occurs when  $\beta = 1.75$ . Through simulation, it has been demonstrated that when the number of receive antennas is 4 and  $\beta = 1.75$  is selected, the performance of the FHT-based decoder approaches that of the ML decoder at a BLER threshold of 1%, corresponding to an additional gain of 1 dB. Furthermore, it is important to note that the adaptive DMRS/data power adjustment process yields greater benefits with an increased number of receiving antennas. Consequently, the disparity between the performance of the ML receiver and the FHT-based receiver can be minimized. Likewise, the simulation outcomes pertaining to a fading channel at the link level, incorporating MIMO, are depicted in Figure 4. The investigation encompasses a MIMO system operating with spatial multiplexing, specifically a  $(4 \times 4)$  configuration. We observe that there exists a performance disparity of 1 dB between the ML receiver and the FHT-based receiver at a BLER of 1%. The adaptive adjustment process allows an additional gain of approximately 1 dB by selecting  $\beta = 1.50$ , thus bridging the gap between the ML and FHT receivers, and 2 dB by selecting  $\beta = 1.75$ , outperforming the ML receiver by 1 dB.

Overall, the implications of performing an adaptive DMRS/data power adjustment within the 3GPP standard are

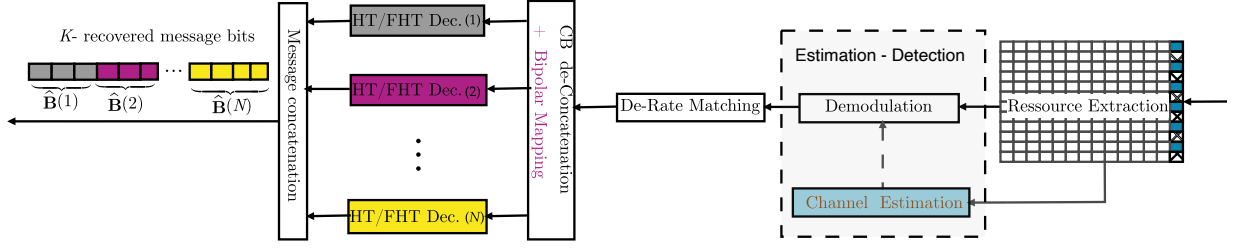


Fig. 2. Block-based Hadamard/Fast Hadamard Transform based decoding: Receiver end.

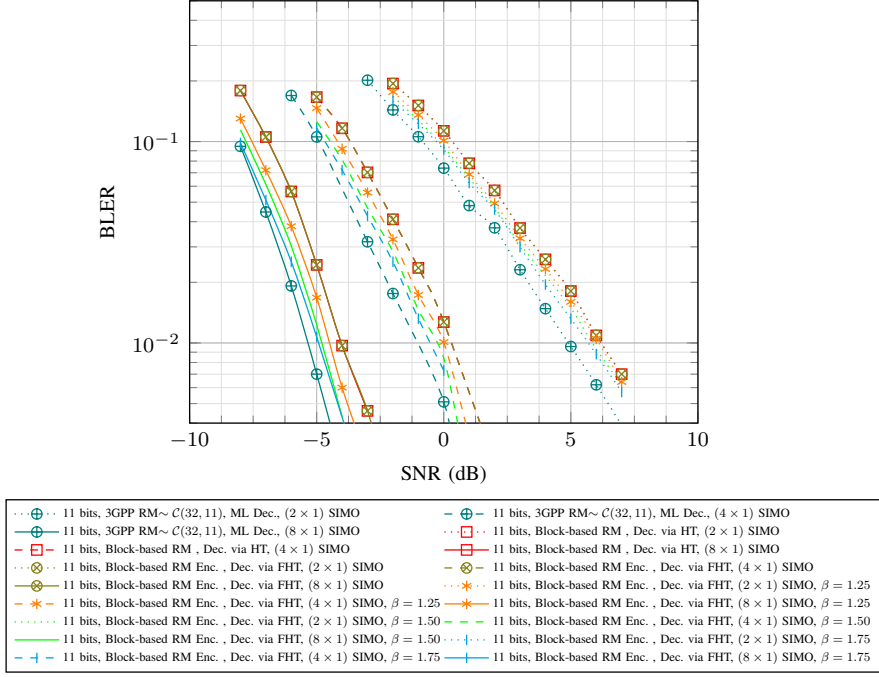


Fig. 3. Block Error Rate, 11 bits, 2 PRB(16 REs=data, 8 REs=DMRS), Block-based decoding via HT & FHT based decoders vs ML decoder, Adaptive DMRS/data Power Adjustment via  $\beta$ ,  $(\{8, 4, 2\} \times 1)$  SIMO, Rayleigh fading Channel, Unknown Channel State Information.

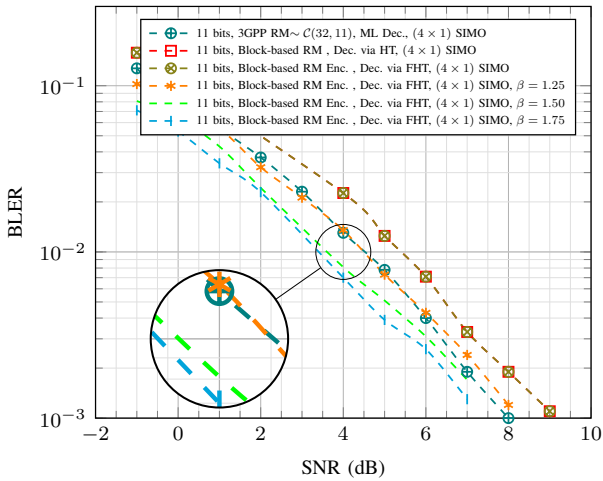


Fig. 4. Block Error Rate, 11 bits, 2 PRB(16 REs=data, 8 REs=DMRS), Block-based decoding via HT & FHT based decoders vs ML decoder, Adaptive DMRS/data Power Adjustment, Unknown Channel State Information (CSI),  $(4 \times 4)$  MIMO, Rayleigh fading Channel.

significant in term of performance improvement. Specifically, the *User equipment* (UE) could vary the power allocation between the DMRS and data. This flexibility in power

allocation is transparent to the receiver.

Furthermore, in terms of computational efficiency, block decoding using the Fast Hadamard transform is more advantageous as it offers a faster convergence time, which we will endeavour to demonstrate hereafter.

The graph in Figure 5 portrays the number of operations required, denoted as  $\mathcal{N}$ , in relation to the input size, denoted as  $N' = 2^K$ , for both the ML decoder and the FHT-based decoders. Notwithstanding the computational time complexity involved can be assessed through comprehensive numerical simulations, we approach this inquiry within a relatively simplified analytical or theoretical framework.

This method seeks to encapsulate the quantum of steps or operations essential for a decoder to achieve convergence and successfully retrieve transmitted messages. To aid comprehension of the graph, we introduce the notations  $N'_1 = 2^4 = 16$  and  $N'_2 = 2^5 = 32$ , representing the number of candidate data words that can be retrieved for the FHT-based decoder 1 and FHT-based decoder 2. These notations correspond to  $\mathcal{N}'_1 = (N'_1 \log N'_1)$  and  $\mathcal{N}'_2 = N'_2 \log N'_2$  operations. As for the ML decoder, which serves to recover encoded codewords using the 3GPP RM code family, the case is such that  $N' = 2^{11} = 2048$ , resulting in  $\mathcal{N}^{(ML)} = N'^2$  operations.

Lastly, focusing on the block-based FHT decoder and initially assuming that decoders are executed in parallel and to some extent independently from each other, as illustrated in Figure 2, this strategy is employed to leverage the merit of parallelism in processing, thereby leading to a diminished inherent complexity. For the block-based FHT with  $N' = N'_1 + N'_2$ , the overall decoding complexity is encapsulated as  $\mathcal{O}(\max(N'_1 \log N'_1, N'_2 \log N'_2))$ , equivalent to  $\mathcal{N}^{(FHT)} = (N'_2 \log N'_2)$ , representing the decoder associated with the longest codeword.

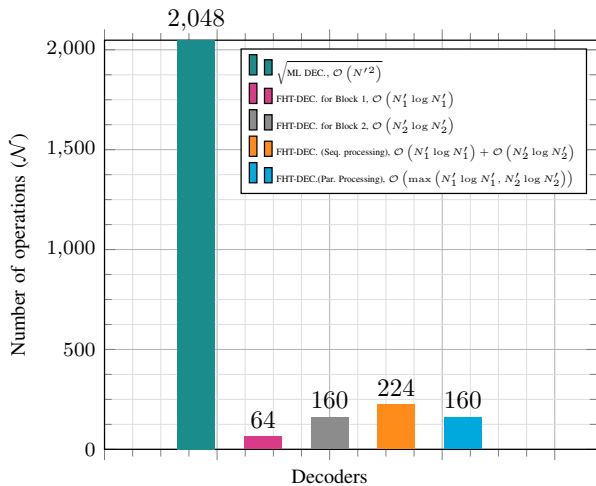


Fig. 5. Comparative analysis, showing the number of  $\mathcal{N}$  operations for the ML decoder compared to block-based FHT-decoders featuring sequential and parallel processing.

The graph distinctly reveals that the ML decoder requires  $2048^2$  operations compared to 160 for the block-based FHT decoder. The number of operations needed by the ML decoder highly surpasses that of the block-FHT decoder even if the FHT-based decoders are executed sequentially. So the merits of decoding using FHT can be visibly perceived in terms of complexity. It is therefore advisable to employ this technique in scenarios where time sensitivity is crucial, particularly in ultra-reliable low-latency communications (URLLC) and industrial IoT applications.

## V. CONCLUSION

This paper presented low-complexity block-based encoding and decoding algorithms for short block length channels. In

terms of the precise use-case, we were primarily concerned with the baseline 3GPP Short block transmissions in which payloads are encoded by Reed-Muller codes and paired with orthogonal DMRS. In contemporary communication systems, the short block decoding often employs the utilization of DMRS-based least squares channel estimation, followed by maximum likelihood decoding. However, it is acknowledged that this approach can incur substantial computational complexity when processing long bit length codes. We proposed an innovative approach to tackle this challenge by introducing the principle of block/segment encoding using First-Order RM Codes which is amenable to low-cost decoding through block-based fast Hadamard transforms. The Block-based FHT has demonstrated to be cost-efficient with regards to decoding time, as it evolves from quadric to quasi-linear complexity with a manageable decline in performance. Additionally, by incorporating an adaptive DMRS/data power adjustment technique, we were able to bridge/reduce the performance gap and attained high sensitivity, leading to a good trade-off between performance and complexity to efficiently handle small payloads.

## REFERENCES

- [1] 3GPP TS 38.212 V16.5.0, "Technical Specification Group Radio Access Network, Multiplexing and channel coding," July 2020.
- [2] I. Dumer and K. Shabunov, "Soft-decision decoding of Reed-Muller codes: recursive lists," in *IEEE Transactions on Information Theory*, vol. 52, no. 3, pp. 1260-1266, March 2006.
- [3] M. Ye and E. Abbe, "Recursive projection-aggregation decoding of Reed-Muller codes," 2019 IEEE International Symposium on Information Theory (ISIT), Paris, France, 2019, pp. 2064-2068.
- [4] J. Li, S. M. Abbas, T. Tonnellier and W. J. Gross, "Reduced Complexity RPA Decoder for Reed-Muller Codes," 2021 11th International Symposium on Topics in Coding (ISTC), Montreal, QC, Canada, 2021, pp. 1-5.
- [5] M. Lian, C. Häger and H. D. Pfister, "Decoding Reed-Muller Codes Using Redundant Code Constraints," 2020 IEEE International Symposium on Information Theory (ISIT), Los Angeles, CA, USA, 2020, pp. 42-47.
- [6] N. Doan, S. A. Hashemi, M. Mondelli and W. J. Gross, "Decoding Reed-Muller Codes With Successive Codeword Permutations," in *IEEE Transactions on Communications*, vol. 70, no. 11, pp. 7134-7145, Nov. 2022.
- [7] Alexei E. Ashikhmin and Simon N. Litsyn, "Fast Decoding Algorithms for First Order Reed-Muller and Related Codes," *Designs, Codes and Cryptography* 7, pp. 187-214, 1996.
- [8] Todd K. Moon, "Error Correction Coding: Mathematical Methods and Algorithms," Wiley-Interscience, 605 Third Avenue New York, NY-United States, May 2005.
- [9] Stephen B. Wicker, "Error Control Systems for Digital Communication and Storage," Prentice Hall, July 1994.