

# Reed–Muller Codes

Hanwen Yao and Henry D. Pfister

January 2025

## 1 Boolean function and polynomial evaluation

A boolean function with  $m$  variables is a mapping from  $\{0, 1\}^m$  to  $\{0, 1\}$ . For example when  $m = 3$ , one such boolean function  $f(x_1, x_2, x_3)$  that maps any realization of three variables  $x_1, x_2, x_3 \in \{0, 1\}$  to a binary value in  $\{0, 1\}$  is specified by the truth table

$x_1$	0	0	0	0	1	1	1	1
$x_2$	0	0	1	1	0	0	1	1
$x_3$	0	1	0	1	0	1	0	1
$f$	0	0	0	1	1	1	1	0

The last row of the table gives the value taken by  $f$ , and is a binary vector of length  $2^m$ . Since the last row can be filled arbitrarily, there are  $2^{2^m}$  boolean functions of  $m$  variables.

A boolean function  $f$  can be written directly from its truth table. In this example,  $f$  can be written as

$$f = (1 - x_1)x_2x_3 + x_1(1 - x_2)(1 - x_3) + x_1(1 - x_2)x_3 + x_1x_2(1 - x_3)$$

which simplifies to

$$f = x_1 + x_2x_3$$

This is a polynomial over  $x_1, x_2, x_3$ . Note that  $x_i^2 = x_i$  in binary. In general, any boolean function can be viewed as a polynomial over  $x_1, x_2, \dots, x_m$ , and it can be written as a sum of monomials

$$1, x_1, x_2, \dots, x_m, x_1x_2, x_1x_3, \dots, x_{m-1}x_m, \dots, x_1x_2 \cdots x_m$$

with coefficients in  $\{0, 1\}$ . Since the number of those monomials equals

$$1 + \binom{m}{1} + \binom{m}{2} + \dots + \binom{m}{m-1} + \binom{m}{m} = 2^m,$$

they are linearly independent.

## 2 Definition of the Reed-Muller code

For a polynomial  $f$  over  $m$  variables  $x_1, x_2, \dots, x_m$ , denote  $\underline{f}$  as the length- $2^m$  vector obtained from the evaluation of  $x_1, x_2, \dots, x_m$  with  $(x_1, x_2, \dots, x_m)$  ranges over  $\{0, 1\}^m$ . For example, when  $f = x_1 + x_2x_3$

$$\underline{f} = 00011110$$

**Definition 1.** The  $r$ -th order binary Reed-Muller code  $\mathcal{R}(r, m)$  of length  $n = 2^m$ , for  $0 \leq r \leq m$ , is the set of evaluations  $\underline{f}$  for all polynomials with degree  $\leq r$ .

**Example 1.** The first order RM code of length  $2^3 = 8$  has the generator matrix:

$$G = \begin{bmatrix} \underline{1} \\ x_1 \\ x_2 \\ x_3 \end{bmatrix} = G = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}$$

Does this look familiar? Hint: this is the parity-check matrix of the (8,4,4) extended Hamming code. In general, the  $r$ -th order Reed-Muller code  $\mathcal{R}(r, m)$  is generated by the evaluations of monomials

$$1, x_1, x_2, \dots, x_m, x_1x_2, \dots, x_{m-1}x_m, \dots \text{ (up to degree } r \text{)}$$

which form a basis for the code. So  $\mathcal{R}(r, m)$  has dimension

$$k = 1 + \binom{m}{1} + \binom{m}{2} + \dots + \binom{m}{r} = \sum_{i=0}^r \binom{m}{i}$$

**Lemma 1.** *The evaluation vector of a single monomial of degree  $r$  has weight  $2^{m-r}$ .*

*Proof.* A monomial of degree  $r$  evaluates to 1 if and only if all involved variables are set to 1. Since  $r$  variables are involved, this occurs for a fraction  $1/2^r$  of the evaluation points. Since there are a total of  $2^m$  evaluation points, the weight of the evaluation vector is given by  $2^{m-r}$ .  $\square$

Remark: This lemma shows that  $d_{\min}(\mathcal{R}(r, m)) \leq 2^{m-r}$ , because that codeword weight can be generated by monomials of degree  $r$ .

**Lemma 2.**  *$\mathcal{R}(m-1, m)$  is the single parity-check code.*

*Proof.* To see that  $\mathcal{R}(m-1, m)$  equals the single parity-check code  $\mathcal{C}_0$  of length  $n = 2^m$ , we first observe that it is spanned by the evaluation vectors of monomials of degree at most  $m-1$ . Lemma 1 shows that the evaluation vector of a monomial of degree  $r$  has weight  $2^{m-r}$  and we note that these weights are even for  $r \in \{0, 1, \dots, m-1\}$ . Thus, all codewords have even weight because all linear combinations of even weight vectors have even weight. This implies that  $\mathcal{R}(m-1, m)$  is a subset of  $\mathcal{C}_0$ . To see that it equals  $\mathcal{C}_0$ , we note that  $\dim(\mathcal{R}(m-1, m)) = n-1$  equals the dimension of  $\mathcal{C}_0$ .  $\square$

**Lemma 3.**  $\mathcal{R}(r, m)^\perp = \mathcal{R}(m-r-1, m)$

*Proof.* Consider the product of two polynomials  $f$  and  $g$  with  $\underline{f} \in \mathcal{R}(r, m)$  and  $\underline{g} \in \mathcal{R}(m-r-1, m)$ , then we have  $\deg(f) \leq r$  and  $\deg(g) \leq m-r-1$ . Since  $h = fg$  has degree at most  $r + (m-r-1) = m-1$ , its evaluation vector belongs to  $\mathcal{R}(m-1, m)$ . So by Lemma 2,  $\underline{h}$  has even weight, which means  $\underline{f}$  and  $\underline{g}$  are orthogonal.

It follows that  $\mathcal{R}(m-r-1, m) \subseteq \mathcal{R}(r, m)^\perp$ . To see that  $\mathcal{R}(m-r-1, m)$  is not larger than  $\mathcal{R}(r, m)^\perp$ , we observe that they have the same dimension,

$$\dim(\mathcal{R}(m-r-1, m)) = \sum_{i=0}^{m-r-1} \binom{m}{i} = n - \sum_{i=0}^r \binom{m}{i} = \dim(\mathcal{R}(r, m)^\perp).$$

This completes the proof.  $\square$

### 3 Plotkin construction

Let  $\mathcal{C}_1$  be an  $(n, k_1, d_1)$  linear code with generator matrix  $G_1$  and parity-check matrix  $H_1$ , and let  $\mathcal{C}_2$  be an  $(n, k_2, d_2)$  linear code with generator matrix  $G_2$  and parity-check matrix  $H_2$ . Then, Plotkin's construction defines a new  $(2n, k, d)$  linear code,

$$\mathcal{C} = \{(\underline{u}, \underline{u} + \underline{v}) \mid \underline{u} \in \mathcal{C}_1, \underline{v} \in \mathcal{C}_2\}$$

with generator and parity-check matrices given by

$$G = \begin{bmatrix} G_1 & G_1 \\ 0 & G_2 \end{bmatrix} \quad H = \begin{bmatrix} H_1 & 0 \\ -H_2 & H_2 \end{bmatrix}.$$

To see that  $G$  is a generator, we note that

$$(\underline{m}_1, \underline{m}_2)G = (\underline{m}_1G_1, \underline{m}_1G_1 + \underline{m}_2G_2) = (\underline{u}, \underline{u} + \underline{v}), \quad \underline{u} \in \mathcal{C}_1, \underline{v} \in \mathcal{C}_2.$$

The existence of a generator also implies that  $\mathcal{C}$  is linear. Subtracting the 1st column from the 2nd column also shows that

$$k = \text{rank}(G) = \text{rank}(G_1) + \text{rank}(G_2) = k_1 + k_2$$

One can verify that  $H$  is a parity-check matrix by computing  $GH^T$  and noting that

$$2n - k = \text{rank}(H) = \text{rank}(H_1) + \text{rank}(H_2) = 2n - k_1 - k_2$$

**Lemma 4.** *If  $\mathcal{C}_1$  has minimum distance  $d_1$  and  $\mathcal{C}_2$  has minimum distance  $d_2$ , then  $\mathcal{C}$  has minimum distance  $d = \min\{2d_1, d_2\}$ .*

*Proof.* Consider a codeword  $(\underline{u}, \underline{u} + \underline{v})$  in  $\mathcal{C}$ . If  $\underline{v} = 0$ , we have

$$\text{wt}(\underline{u}, \underline{u}) = 2\text{wt}(\underline{u}) \geq 2d_1,$$

where the equality is achievable by picking  $\underline{u}$  to be the min-weight codeword in  $\mathcal{C}_1$ . And if  $\underline{v} \neq 0$ , we have

$$\begin{aligned} \text{wt}(\underline{u}, \underline{u} + \underline{v}) &= \text{wt}(\underline{u}) + \text{wt}(\underline{u} + \underline{v}) \\ &\geq \text{wt}(\underline{u} + \underline{u} + \underline{v}) \\ &= \text{wt}(\underline{v}) \\ &\geq d_2, \end{aligned}$$

where the equality is also achievable by picking  $\underline{u} = 0$  and  $\underline{v}$  to be the min-weight codeword in  $\mathcal{C}_2$ . Therefore, we have  $d = \min\{2d_1, d_2\}$ .  $\square$

**Example 2.** *Let  $\mathcal{C}_1 = \{00, 01, 10, 11\}$  and  $\mathcal{C}_2 = \{00, 11\}$ . Then,*

$$\mathcal{C} = \{0000, 0101, 1010, 1111, 0011, 0110, 1001, 1100\}$$

*has  $k = k_1 + k_2 = 2 + 1 = 3$  and  $d = \min(2, 2) = 2$ .*

Now we show that Reed-Muller codes of length  $2^{m+1}$  can be obtained from Reed-Muller codes of length  $2^m$  using the Plotkin construction.

**Lemma 5.**

$$\mathcal{R}(r+1, m+1) = \{(\underline{u}, \underline{u} + \underline{v}) \mid \underline{u} \in \mathcal{R}(r+1, m), \underline{v} \in \mathcal{R}(r, m)\}$$

*Proof.* By definition a codeword in  $\mathcal{R}(r+1, m+1)$  is the evaluation of a polynomial  $f(x_1, \dots, x_m, x_{m+1})$  with degree  $\leq r+1$ . Then we can write  $f$  as

$$f(x_1, \dots, x_m, x_{m+1}) = g(x_1, \dots, x_m) + x_{m+1}h(x_1, \dots, x_m),$$

where  $\deg(g) \leq r+1$  and  $\deg(h) \leq r$ . Note that the polynomials  $g$  and  $h$  are unique in this representation. Let  $\underline{g}$  and  $\underline{h}$  be the evaluation vectors of  $g$  and  $h$  of length  $2^m$ , then the evaluation vectors of  $g$  and  $h$  with  $(x_1, x_2, \dots, x_{m+1})$  ranges over  $\{0, 1\}^{m+1}$  are  $(\underline{g}, \underline{g})$  and  $(0, \underline{h})$ , respectively. Hence

$$\underline{f} = (\underline{g}, \underline{g}) + (0, \underline{h}) = (\underline{g}, \underline{g} + \underline{h})$$

with  $\underline{g} \in \mathcal{R}(r+1, m)$  and  $\underline{h} \in \mathcal{R}(r, m)$ .  $\square$

If we check the dimensions of those Reed-Muller codes, we have

$$\binom{m+1}{r+1} = \binom{m}{r+1} + \binom{m}{r},$$

which is exactly the recursive formula for the binomial coefficients.

**Lemma 6.** *The minimum distance of  $\mathcal{R}(r, m)$  is given by  $d(r, m) = 2^{m-r}$ .*

*Proof.* First, it is easy to verify that  $\mathcal{R}(0, m)$  is the repeat by  $2^m$  code with  $d(0, m) = 2^m$ . And  $\mathcal{R}(m, m)$  is the entire space of  $\{0, 1\}^{2^m}$  with  $d(m, m) = 1$ . This completes the proof for  $m = 1$ . Next, we proceed by induction on  $m$ . Since

$$\mathcal{R}(r+1, m+1) = \{(\underline{u}, \underline{u} + \underline{v}) \mid \underline{u} \in \mathcal{R}(r+1, m), \underline{v} \in \mathcal{R}(r, m)\}$$

by Lemma 5, applying Lemma 4 shows that

$$d(r+1, m+1) = \min\{2d(r+1, m), d(r, m)\} = 2^{m-r}$$

for  $r \in \{0, 1, \dots, m-1\}$ . The values of  $d(r, m+1)$  for  $r = 0$  and  $r = m+1$  are treated separately using the first argument. This completes the induction.  $\square$

Note that for a family of Reed-Muller codes with rates approaching  $R > 0$ , we have  $r \rightarrow \infty$ , which means their relative distances

$$\delta = \frac{2^{m-r}}{2^m} = 2^{-r} \rightarrow 0.$$

However, in the stochastic noise model, it has been shown that Reed-Muller codes actually achieve the capacity on both BEC [1] and BSC [2, 3].

## 4 Permutation Automorphism

Let  $S_n$  denote the symmetric group on  $n$  elements. For  $\sigma \in S_n$  and  $\underline{x} \in \{0, 1\}^n$ , let

$$\underline{x}_\sigma = (x_{\sigma(1)}, x_{\sigma(2)}, \dots, x_{\sigma(n)})$$

**Definition 2.** *Two codes  $\mathcal{C}$  and  $\mathcal{C}'$  of length  $n$  are called equivalent if there is a permutation  $\sigma \in S_n$  such that  $\underline{x}_\sigma = (x_{\sigma(1)}, x_{\sigma(2)}, \dots, x_{\sigma(n)}) \in \mathcal{C}'$  for all  $\underline{x} \in \mathcal{C}$ .*

**Definition 3.** *The permutation automorphism group of  $\mathcal{C}$  is defined to be*

$$\text{Per}(\mathcal{C}) \triangleq \{\sigma \in S_n \mid \underline{x}_\sigma \in \mathcal{C}, \forall \underline{x} \in \mathcal{C}\}$$

These permutations form a subgroup of  $S_n$  because, if two permutations separately preserve the set of codewords, then their composition also preserves the set of codewords. Thus, the set of permutations in  $\text{Per}(\mathcal{C})$  is closed under composition and must form a subgroup.

**Example 3.** *Consider the  $(4, 2, 2)$  linear code with codewords*

$$\mathcal{C} = \left\{ \begin{array}{l} 0000 \\ 1100 \\ 0011 \\ 1111 \end{array} \right\}.$$

*Observe that swapping  $x_1 \leftrightarrow x_2$  or  $x_3 \leftrightarrow x_4$  leaves the list of codewords unchanged. Also, swapping  $x_1x_2 \leftrightarrow x_3x_4$  changes the list but the entire set of codewords remains unchanged. Thus, all of these permutations are automorphisms.*

For linear codes, the permutation automorphism group satisfies several properties.

**Lemma 7.**  *$\text{Per}(\mathcal{C}) = \text{Per}(\mathcal{C}^\perp)$*

*Proof.* For all  $\sigma \in \text{Per}(\mathcal{C})$ ,  $\underline{x} \in \mathcal{C}$ , and  $\underline{y} \in \mathcal{C}^\perp$ , Definition 3 implies that

$$\sum_{i=1}^n x_{\sigma(i)} y_i = 0.$$

Changing the variable of summation to  $j = \sigma(i)$  shows that

$$\sum_{j=1}^n x_j y_{\sigma^{-1}(j)} = 0.$$

Thus,  $\sigma^{-1} \in \text{Per}(\mathcal{C}^\perp)$  because  $\underline{y}_{\sigma^{-1}} \in \mathcal{C}^\perp$  for all  $\sigma \in \text{Per}(\mathcal{C})$  and  $\underline{y} \in \mathcal{C}^\perp$ . Since every element in a group has a unique inverse in the group, this implies that  $\text{Per}(\mathcal{C}) \subseteq \text{Per}(\mathcal{C}^\perp)$ . Also, a symmetric argument holds starting from  $\sigma \in \text{Per}(\mathcal{C}^\perp)$ , so we see that  $\text{Per}(\mathcal{C}^\perp) \subseteq \text{Per}(\mathcal{C})$ . Hence, the two groups are equal.  $\square$

**Theorem 1.** *For an  $(n, k)$  binary linear code  $\mathcal{C}$ , we have  $\sigma \in \text{Per}(\mathcal{C})$  if and only if there is an invertible  $k \times k$  matrix  $L$  over  $\mathbb{F}_2$  such that  $LG_\sigma = G$ , where  $G$  is a generator matrix for  $\mathcal{C}$ . Similarly, we have  $\sigma \in \text{Per}(\mathcal{C})$  if and only if there is an invertible  $(n - k) \times (n - k)$  matrix  $L'$  over  $\mathbb{F}_2$  such that  $L'H_\sigma = H$ , where  $H$  is a parity-check matrix for  $\mathcal{C}$ .*

*Proof.* Since a binary linear code  $\mathcal{C}$  is a  $k$ -dimensional subspace of  $\mathbb{F}_2^n$ , a matrix  $G'$  is a generator for  $\mathcal{C}$  if and only if its rows form a basis for  $\mathcal{C}$  (i.e., it can be transformed by left-multiplication into a known generator for  $\mathcal{C}$ ). Thus, the matrix  $G_\sigma$  is a generator for  $\mathcal{C}$  if and only if there is an invertible  $k \times k$  matrix  $L$  satisfying  $LG_\sigma = G$ . For the case of parity-check matrices, note that its parity-check matrix generates the dual code. Thus, for any  $\sigma \in \text{Per}(\mathcal{C}^\perp)$ , there is an invertible  $(n - k) \times (n - k)$  matrix  $L$  satisfying  $LH_\sigma = H$ . Since Lemma 7 shows that  $\text{Per}(\mathcal{C}) = \text{Per}(\mathcal{C}^\perp)$ , the same result also holds for all  $\sigma \in \text{Per}(\mathcal{C})$ .  $\square$

**Corollary 1** (Optional). *For a binary linear code  $\mathcal{C}$ , the permutation group  $\text{Per}(\mathcal{C})$  is isomorphic to some subgroup of  $GL_m(\mathbb{F}_2)$  (i.e., the general linear group of invertible  $m \times m$  matrices over  $\mathbb{F}_2$ ) where  $m = \min\{k, n - k\}$ .*

*Proof.* Without loss of generality, assume that  $k \leq n - k$ . For each  $\sigma \in \text{Per}(\mathcal{C})$ , there is an invertible  $m \times m$  matrix  $L$  that induces the permutation  $\sigma$  on the columns of the generator  $G$ . Thus, we can represent the composition of elements  $\sigma, \sigma' \in \text{Per}(\mathcal{C})$  by the product of invertible  $m \times m$  matrices  $L, L'$  over  $\mathbb{F}_q$ .

In particular, we can read off the permutation  $\sigma'(\sigma(\cdot))$  associated with  $L'L$  by computing  $G' = L'LG$  and then matching the columns with  $G$ . If all columns are unique (i.e., the minimum distance is at least 3), then the permutation will be specified uniquely. Otherwise, there will be trivial automorphisms that interchange identical columns and leave the information sequence unchanged.  $\square$

## 5 Automorphism group of Reed-Muller code

Let  $A = (a_{ij})$  be an invertible  $m \times m$  binary matrix and let  $\underline{b} \in \{0, 1\}^m$  be a vertical binary vector of length  $m$ , then the mapping

$$T : \underline{x} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{pmatrix} \mapsto A\underline{x} + \underline{b} = A \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{pmatrix} + \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{pmatrix}$$

is called an *affine transformation*, and it defines a bijection on  $\{0, 1\}^m$ .

Consider the evaluation vector of a polynomial  $f$  over  $m$  variables, where we use  $\underline{x} = (x_1, x_2, \dots, x_m)^T$  as the index for the bit  $f(x_1, x_2, \dots, x_m)$  in  $\underline{f}$  for all  $\underline{x} \in \{0, 1\}^m$ . The mapping  $T$  defines the following permutation

$$\sigma \in S_{2^m} : \{0, 1\}^m \rightarrow \{0, 1\}^m, \quad \sigma(\underline{x}) \mapsto A\underline{x} + \underline{b}$$

on the coordinates of  $f$ . A moment's thought reveals that after this permutation, we obtain the evaluation vector of the new polynomial

$$Tf = f \left( \sum_j a_{1j}x_j + b_1, \sum_j a_{2j}x_j + b_2, \dots, \sum_j a_{mj}x_j + b_m \right)$$

**Example 4.** Consider the polynomial  $f(x_1, x_2, x_3) = x_1 + x_2x_3$  with

$$\begin{array}{c|cccccccc} x_1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ x_2 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ x_3 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ \hline f & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \end{array},$$

and the following permutation on the evaluation of  $f$ :

$$T : \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} \mapsto \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$

Then after the permutation, we obtain the evaluation vector of the new polynomial

$$Tf = f(x_1 + x_2, x_2, x_3 + 1) = (x_1 + x_2) + x_2(x_3 + 1) = 1 + x_1 + x_2 + x_2x_3$$

It is clear that if  $f$  is a polynomial of degree  $r$ , so is  $Tf$ . Therefore  $T$  belongs to the permutation automorphism group of Reed-Muller code  $\mathcal{R}(r, m)$ . Denote  $GA(m)$  as the group of all the affine transformations on  $m$  variables, then we know

$$GA(m) \subseteq \text{Per}(\mathcal{R}(r, m)).$$

In fact, it has been shown that

$$GA(m) = \text{Per}(\mathcal{R}(r, m)).$$

**Definition 4.** A permutation group  $\mathcal{G} \subseteq S_n$  is called transitive if, for all  $i, j \in [n]$ , there is a permutation  $\sigma \in \mathcal{G}$  such that  $\sigma(i) = j$ . It is called doubly transitive if, for all  $i, j, k, l \in [n]$  such that  $i \neq j$  and  $k \neq l$ , there is a permutation  $\sigma \in \mathcal{G}$  such that  $\sigma(i) = k$  and  $\sigma(j) = l$ .

Remark: For a code  $\mathcal{C}$  on a memoryless channel where  $\text{Per}(\mathcal{C})$  is transitive, the probability of symbol error under ML decoding is the same for all symbols. For a code  $\mathcal{C}$  where  $\text{Per}(\mathcal{C})$  is doubly transitive, all pairs of symbols have the same joint distribution of errors.

Finally, the affine group  $GA(m)$  is doubly transitive because, for any  $\underline{w}, \underline{x}, \underline{y}, \underline{z} \in \{0, 1\}^m$  such that  $\underline{w} \neq \underline{x}$  and  $\underline{y} \neq \underline{z}$ , one can find an invertible  $A$  with a  $\underline{b} \in \{0, 1\}^m$  such that,  $\underline{y} = A\underline{w} + \underline{b}$  and  $\underline{z} = A\underline{x} + \underline{b}$ .

**Theorem 2** ([1]). Let  $\mathcal{C}_m$  be a sequence of binary codes with increasing blocklengths whose rates converge to  $R \in (0, 1)$ . If these codes are transmitted over a BEC( $\epsilon$ ), with  $\epsilon < 1 - R$ , and  $\text{Per}(\mathcal{C}_m)$  is doubly transitive for all  $m$ , then the bit erasure rate of optimal decoding converges to 0 as  $m \rightarrow \infty$ . Thus, this code sequence achieves capacity.

**Corollary 2** ([1]). A sequence  $\mathcal{C}_m = \mathcal{R}(r_m, m)$  of Reed-Muller codes achieves capacity on the BEC if the implied rate sequence  $R_m$  converges to some  $R \in (0, 1)$ .

## References

- [1] S. Kudekar, S. Kumar, M. Mondelli, H. D. Pfister, E. Şaşoğlu, and R. Urbanke, "Reed-Muller codes achieve capacity on erasure channels," in *Proc. of the Annual ACM Symp. on Theory of Comp.*, 2016.
- [2] G. Reeves and H. D. Pfister, "Reed-Muller codes on BMS channels achieve vanishing bit-error probability for all rates below capacity," *IEEE Trans. Inform. Theory*, 2023.
- [3] E. Abbe and C. Sandon, "A proof that Reed-Muller codes achieve Shannon capacity on symmetric channels," in *Proc. IEEE Symp. on the Found. of Comp. Sci.*, 2023, pp. 177–193.