On the Equivalence between Pre-transformed and Parity-check Monomial Codes

Yuanxin Guo*, Zihan Tang†, Bin Li†
*Wireless Technology Laboratory, Huawei Technologies, Shenzhen, P. R. China
School of Science and Engineering, the Chinese University of Hong Kong, Shenzhen, P. R. China
yuanxinguo@link.cuhk.edu.cn
†Wireless Technology Laboratory, Huawei Technologies, Shenzhen, P. R. China
{tangzihan1, binli.binli}@huawei.com

***Abstract*—Polar codes and Reed-Muller codes belong to a family of codes called monomial codes. In this work, we study pre-transformed monomial codes, which cover several constructions including parity-check (PC) codes and PAC codes. We show that any pre-transformed monomial code can be transformed into a parity-check monomial code with the same codewords, and give an explicit algorithm for this transformation. We further prove that for certain monomial codes, the minimum weight is invariant under pre-transformation, but specific pre-transformation matrices can be constructed to reduce the number of minimum-weight codewords. These results offer theoretical support for the success of various heuristics, e.g., PAC codes attain dispersion bound, and provide guidance for designing short codes.**

***Index Terms*—Polar codes, RM codes, code distance.**

I. Introduction

Polar codes [1] are the first explicitly constructed codes that achieve channel capacity. The code construction is tailored for an efficient successive cancellation (SC) decoder. However, the performance of polar codes under SC decoding is inferior to LDPC or turbo codes. A successive cancellation list (SCL) decoding algorithm was proposed in [2], whose performance approaches maximum likelihood (ML) decoding as the list size L increases. Nonetheless, the performance of ML decoding is undermined by the small code distance of polar codes.

Several methods have been proposed to improve the weight spectrum of polar codes, e.g., eBCH polar subcodes [3] and LWB-polar codes [4]. In addition, a class of such constructions, including CRC-aided (CA) polar codes [5], RM-polar codes [6], PC-polar codes [7], and PAC codes [8], can be fit in a unifying framework called *pre-transformed polar codes*, which was proposed in [9]. In [9], it was shown that pre-transformation does not reduce code distance. A recursive formula was proposed in [10] to calculate the average weight spectrum of pre-transformed polar codes.

In general, the method of pre-transformation can be extended to a larger family of codes having similar structure to polar codes, which are called monomial codes. Our subsequent discussion is aimed for general pre-transformed monomial codes, where we first inspect the structure of these codes, and then derive properties of their weight spectra. Our main contributions include:

- Prove that all pre-transformed monomial codes, where PAC codes are special instances, can be regarded as instances of PC monomial codes.
- Present an algorithm that constructs a PC monomial code having the same codewords as any given pre-transformed monomial code.
- Prove that pre-transformation does not change the code distance for a subclass of monomial codes containing RM codes and polar codes.
- Construct pre-transformation matrices that strictly reduce the number of minimum-weight codewords for monomial codes satisfying certain assumptions.

The paper is organized as follows. In Section II, we review the basics of Reed-Muller codes, polar codes, and monomial codes. Pre-transformed monomial codes are introduced in the end of Section II and we prove in Section III that these codes are equivalent to PC monomial codes, where an algorithm is given to convert a pre-transformed monomial code to a PC monomial code. In Section IV, we present some new results on the weight spectrum of pre-transformed monomial codes.

II. Preliminaries

A. Reed-Muller (RM) Codes

Let $\boldsymbol{F} = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$ be a binary matrix, and let the $N \times N$ matrix $\boldsymbol{H}_N = \boldsymbol{F}^{\otimes n}$ be the nth Kronecker power of matrix $\boldsymbol{F}$, where $N = 2^n$, and $\boldsymbol{F}^{\otimes n} = \boldsymbol{F} \otimes \boldsymbol{F}^{\otimes(n-1)}$. From [1, Proposition 17], the weight of the ith row of $\boldsymbol{H}_N$ is

$$\mathrm{wt}(\boldsymbol{h}_N^{(i)}) = 2^{\mathrm{wt}((i-1)_2)},\ \forall 1 \le i \le N$$

where $\mathrm{wt}(\cdot)$ is the Hamming weight of a tuple or a vector, and $(\cdot)_2$ is the binary expansion of an integer.

The Reed-Muller code $\mathcal{R}(n, r)$ is given by

$$\mathcal{R}(n, r) := \{\boldsymbol{c} = \boldsymbol{u}\boldsymbol{H}_N : u_j = 0 \text{ if } j \notin \mathcal{J}_r\},$$

where $\mathcal{J}_r = \{i \in [N] : \mathrm{wt}((i-1)_2) \ge n - r\} \subset [N]$, with $[N]$ being the set of positive integers not greater than N. Otherwise put, the codewords are all possible linear combinations of rows in $\boldsymbol{H}_N$ with row indices in $\mathcal{J}_r$. This code has length N, dimension $\sum_{\ell=0}^{r} \binom{n}{\ell}$, and minimum distance 2^{n-r} [11].

B. Polar Codes

For binary memoryless symmetric (BMS) channels, an (N, k)-polar code $\mathcal{C}$ can be constructed in a similar fashion to RM codes [1], with

$$\mathcal{C} := \{\boldsymbol{c} = \boldsymbol{u}\boldsymbol{H}_N : u_j = 0 \text{ if } j \notin \mathcal{I}\},$$

where $\mathcal{I}$ contains the indices of the k-most reliable bit sub-channels under SC decoding. We refer to $\mathcal{I}$ as *information set* and its complement $\mathcal{I}^C := [N] \setminus \mathcal{I}$ as *frozen set*.

C. Monomial codes

Monomial codes form a large family which contains Reed-Muller codes and polar codes [12]. A monomial in n variables $\{x_0, \cdots, x_{n-1}\}$ over $\mathbb{F}_2$ takes the form

$$\textstyle\prod_{i=0}^{n-1} x_i^{r_i},\ r_i \in \{0, 1\}, \forall\, 0 \leq i \leq n-1$$

since $x^2 \equiv x$ for all $x \in \mathbb{F}_2$. We denote the set of all such monomials by $\mathcal{M}_n$. The general form of a monomial code is

$$\mathcal{C}(\mathcal{F}) = \text{span}\{\mathsf{ev}(f) : f \in \mathcal{F}\},\ \text{for some } \mathcal{F} \subset \mathcal{M}_n,$$

where $\mathsf{ev}(f) := (f(\boldsymbol{u}))_{\boldsymbol{u}\in\mathbb{F}_2^n}$ is the *evaluation vector* of polynomial f, obtained by evaluating f at every point in $\mathbb{F}_2^n$. By independence of monomials, $\dim(\mathcal{C}(\mathcal{F})) = |\mathcal{F}|$.

It is shown in [12] that the evaluation vectors of monomials in $\mathcal{M}_n$ correspond exactly to the rows in $\boldsymbol{H}_N$:

$$\boldsymbol{h}_N^{(i)} = \mathsf{ev}\left(\textstyle\prod_{j=0}^{n-1} x_j^{i_j}\right),\ \text{where } N - i = \sum_{j=0}^{n-1} i_j 2^j.$$

Denote $\prod_{j=0}^{n-1} x_j^{i_j}$ by $f^{(i)}$, any code of the form $\mathcal{C} := \{\boldsymbol{c} = \boldsymbol{u}\boldsymbol{H}_N : u_j = 0 \text{ if } j \notin \mathcal{I}\}$ with $\mathcal{I} \subset [N]$ can be seen as a monomial code, generated by $\mathcal{F} = \{f^{(i)} : i \in \mathcal{I}\}$. The following proposition is thus immediate.

Lemma 1. *RM codes and polar codes are monomial codes.*

D. Pre-transformed Monomial Codes

Given length N monomial code $\mathcal{C}$, we can transform the code via a binary *pre-transformation (PT) matrix* $\boldsymbol{T}$ [9], where

$$\mathcal{C}_T := \{\boldsymbol{c}' = \boldsymbol{u}\boldsymbol{T}\boldsymbol{H}_N : u_j = 0 \text{ if } j \notin \mathcal{I}\}$$

is the pre-transformed code by $\boldsymbol{T}$. Generally, a PT matrix should satisfy two properties: (i) $\boldsymbol{T}$ is upper-triangular; (ii) $T_{i,i} = 1$, $\forall\, i \in [N]$. We denote the collection of all $N \times N$ PT matrices by $\mathcal{T}_N$.

When $\boldsymbol{T}$ is the $N \times N$ identity matrix, the pre-transformation is trivial in the sense that $\mathcal{C}_T = \mathcal{C}$. The pre-transformed monomial codes cover a large family of modified polar codes, which includes the concatenation of PC with polar codes [7], and the recently proposed PAC codes [8].

III. Equivalence between pre-transformed monomial codes and parity-check monomial codes

In this section, we investigate the family of codes obtained by pre-transforming a given monomial code $\mathcal{C}$. We prove that the family is equivalent to the code family obtained by adding parity-checks to $\mathcal{C}$, suggesting some 'good' codes in practice, e.g., PAC codes [8] are both special cases of PC polar codes. We further provide an explicit algorithm for converting a pre-transformed monomial code to a parity-check (PC) monomial code with the same set of codewords. We first briefly introduce PC monomial codes.

A parity-check monomial code is another way of transforming monomial codes [7]. In particular, in a bit sequence $\boldsymbol{u}$, some frozen bits are changed to parity-check (PC) bits such that for parity bit with index j, a PC function of the form

$$u_j \oplus \textstyle\bigoplus_{i\in\mathcal{I}_j} u_i = 0$$

is to be satisfied for some *parity-check set* $\mathcal{I}_j \subset ([j-1] \cap \mathcal{I})$. Specially, we can regard frozen bits as special PC bits, where for frozen bit j, $\mathcal{I}_j = \emptyset$. Generally, a PC monomial code can be written in the following form:

$$\mathcal{C} := \{\boldsymbol{c} = \boldsymbol{u}\boldsymbol{H}_N : u_j \oplus \textstyle\bigoplus_{i\in\mathcal{I}_j} u_i = 0 \text{ if } j \notin \mathcal{I}\},$$

Parity-check monomial codes can be equally interpreted as pre-transformed monomial codes, where the PT matrix $\boldsymbol{T}$ can be constructed such that $T_{i,j} = 1$ iff $i = j$ or $(i, j) \in \mathcal{I}_j \times \mathcal{I}^C$. Since the PC function for any $j \in \mathcal{I}^C$ can be written as

$$u_j = \textstyle\bigoplus_{i\in\mathcal{I}_j} u_i = \bigoplus_{i\in\mathcal{I}_j} v_i = v_j \oplus \bigoplus_{i\in\mathcal{I}_j} v_i,$$

where $\boldsymbol{u} = \boldsymbol{v}\boldsymbol{T}$, $\boldsymbol{v} \in V$, where V is defined to be the vector subspace of all valid bit sequences:

$$V = \{\boldsymbol{v} \in \mathbb{F}_2^n : v_j = 0 \text{ if } j \notin \mathcal{I}\}.$$

Motivated by the structure of the PT matrices constructed from PC monomial codes, we make the following definition:

Definition 1 (Parity-check-transform matrix)**.** Given monomial code $\mathcal{C}$ with information set $\mathcal{I}$, $\boldsymbol{S}$ is an $N \times N$ *parity-check-transform (PCT) matrix* with respect to code $\mathcal{C}$ if

1) $\boldsymbol{S} \in \mathcal{T}_N$, i.e., $\boldsymbol{S}$ upper-triangular and $S_{i,i} = 1, \forall\, i \in [N]$.
2) For $1 \leq i < j \leq N$, $S_{i,j} = 1$ only if $i \in \mathcal{I}$, $j \in \mathcal{I}^C$.

We denote the collection of $N \times N$ parity-check-transform (PCT) matrices with respect to code $\mathcal{C}$ by $\mathcal{S}_\mathcal{C}$.

We now prove the main result of this section.

Theorem 2. *Let $\mathcal{C}$ be a length-N monomial code with information set $\mathcal{I}$. For any PT matrix $\boldsymbol{T} \in \mathcal{T}_N$, there exists a PCT matrix $\boldsymbol{S} \in \mathcal{S}_\mathcal{C}$ such that $\mathcal{C}_T = \mathcal{C}_S$.*

Proof. Let $\boldsymbol{U} = \boldsymbol{T}$ and $\boldsymbol{T}' = \boldsymbol{T}$, where we in addition require $U_{i,j} = 0$ if $i < j, i, j \in \mathcal{I}^C$ and $T'_{i,j} = 0$ if $i < j, i \in \mathcal{I}^C$. Note that $\boldsymbol{U}$ is invertible with upper-triangular inverse. Let $\boldsymbol{S} = \boldsymbol{U}^{-1}\boldsymbol{T}'$, we claim that $\boldsymbol{S}$ is the PT matrix we desire.

It is not difficult to observe that $\boldsymbol{S}$ is upper-triangular since both $\boldsymbol{U}^{-1}$ and $\boldsymbol{T}'$ are upper-triangular. We now check $S_{i,i} = 1$. Note that since $\boldsymbol{U}\boldsymbol{S} = \boldsymbol{T}'$ we have

$$\bigoplus_{\ell=1}^{N} U_{i,\ell} S_{\ell,i} = U_{i,i} S_{i,i} = T''_{i,i} = T_{i,i} = 1$$

for all $i \in [N]$, Since $U_{i,i} = 1$ by construction, we have $S_{i,i} = 1$ for all $i \in [N]$.

We now show that $\boldsymbol{S} \in \mathcal{S}_{\mathcal{C}}$. It suffices to show that whenever $i < j$, either $i \in \mathcal{I}^C$ or $j \in \mathcal{I}$ implies $S_{i,j} = 0$. We use the notation $\overrightarrow{\boldsymbol{m}}^{(j)}$ to represent the j-th column of matrix $\boldsymbol{M}$.

• $S_{i,j} = 0$ for all $i < j$ and $j \in \mathcal{I}$. Note that by construction, we have $\overrightarrow{\boldsymbol{u}}^{(j)} = \overrightarrow{\boldsymbol{t}'}^{(j)}, \ \forall\, j \in \mathcal{I}$ and also

$$\bigoplus_{i=1}^{N} S_{i,j} \cdot \overrightarrow{\boldsymbol{u}}^{(i)} = \overrightarrow{\boldsymbol{t}'}^{(j)},$$

where $\overrightarrow{\boldsymbol{u}}^{(i)}$ are linearly independent, which implies $S_{j,j} = 1$ is the unique nonzero entry in the jth column.

• $S_{i,j} = 0$ for all $i < j, i, j \notin \mathcal{I}$. Note that by construction, we have $\boldsymbol{u}^{(i)} = \boldsymbol{e}_i, \ \forall\, i \notin \mathcal{I}$, where $\boldsymbol{u}^{(i)}$ denotes the ith row of $\boldsymbol{U}$. Hence for any $i \notin \mathcal{I}$, $U_{i,\ell} = 1$ if and only if $\ell = i$. Note that $T'_{i,j} = 0$ for all $i < j, i, j \notin \mathcal{I}$, then for any $j \notin \mathcal{I}$,

$$\bigoplus_{\ell=1}^{N} S_{\ell,j} \cdot U_{i,\ell} = S_{i,j} U_{i,i} = T'_{i,j},$$

implying $S_{i,j} = 0$ for all $i < j$, $i, j \notin \mathcal{I}$.

Finally, it remains to show that $\mathcal{C}_T = \mathcal{C}_S$. Recall the definition of the vector subspace V before Definition 1. By the definition of pre-transformed monomial codes, it suffices to show $V\boldsymbol{T} = V\boldsymbol{S}$. We first show $V\boldsymbol{T} = V\boldsymbol{T}'$. This is obvious since T and T' only differ in the rows whose indices are in $\mathcal{I}^C$, yet $v_j = 0$ for $j \in \mathcal{I}^C$. Now we show $V\boldsymbol{S} = V\boldsymbol{T}' = V\boldsymbol{U}\boldsymbol{S}$. It suffices to show $V = V\boldsymbol{U}$. Since U is invertible, it suffices to show $V\boldsymbol{U} \subset V$. Note that for any $\boldsymbol{v} \in V$, we have

$$\boldsymbol{v}\boldsymbol{U} = \bigoplus_{i \in \mathcal{I}} v_i \boldsymbol{u}^{(i)} \in V,$$

thus finishing the proof. □

The following is an explicit algorithm that constructs an PCT matrix $\boldsymbol{S}$ that pre-transforms $\mathcal{C}$ to the identical code $\mathcal{C}_T$ for any PT matrix $\boldsymbol{T}$.

Algorithm 1 PCT MATRIX CONSTRUCTION

Require: Pre-transformation matrix $\boldsymbol{T} \in \mathcal{T}_N$.
Ensure: Parity-check-transformation matrix $\boldsymbol{S} \in \mathcal{S}_{\mathcal{C}}$.
1: **for** $i \in [N] \setminus \mathcal{I}$ **do**
2: $\quad \boldsymbol{t}^{(i)} \leftarrow \boldsymbol{e}_i$
3: **end for**
4: Initialize $\boldsymbol{U} \leftarrow \boldsymbol{I}_N$
5: **for** $i \in \mathcal{I}$ **do**
6: $\quad \boldsymbol{u}^{(i)} \leftarrow \boldsymbol{t}^{(i)} \wedge (j \in \mathcal{I})_{j=1}^{n}$ (logical AND of $\boldsymbol{t}^{(i)}$ with the logical vector whose jth entry is given by the boolean value of $j \in \mathcal{I}$)
7: **end for**
8: $\boldsymbol{S} = \boldsymbol{U} \setminus \boldsymbol{T}$ (MATLAB notation, solves $\boldsymbol{U}\boldsymbol{S} = \boldsymbol{T}$)

Algorithm 1 and the discussion prior to Theorem 2 together imply that any pre-transformed monomial code $\mathcal{C}_T$ can be regarded as a parity-check monomial code with PC functions

$$u_j \oplus \bigoplus_{i \in \mathcal{S}_j} u_i = 0, \ \forall\, j \notin \mathcal{I}, \ \text{where } \mathcal{S}_j := \{i \in \mathcal{I}, S_{i,j} = 1\}.$$

and $\boldsymbol{S}$ is the output of Algorithm 1 when the input is $\boldsymbol{T}$. The following corollary is direct.

Corollary 3. *Pre-transformed monomial codes are equivalent to parity-check monomial codes.*

We close this section by an example illustrating that the size of the PCT matrix family is usually tiny compared to that of PT matrix family. This shows our result is significant in terms of designing a pre-transformation matrix as it largely reduces the size of searching space.

Example 1. Consider the second-order $(32, 16)$-RM codes $\mathcal{R}(5, 2)$. The size of the PT matrix family is $|\mathcal{T}| = 2^{496}$, whereas the size of the PCT matrix family is only $|\mathcal{S}_{\mathcal{C}}| = 2^{35}$.

IV. WEIGHT STRUCTURE OF PRE-TRANSFORMED MONOMIAL CODES

In this section, we present two results regarding the weight spectrum of pre-transformed monomial codes. The first is on the code distance, where we enhance the result in [9] and prove that for decreasing monomial codes [12], the minimum distance is invariant under pre-transformation. The second is on the number of minimum-weight codewords, where we prove that for decreasing monomial codes satisfying certain assumptions, a PCT matrix can be constructed to reduce the number of minimum weight codewords. The results suggest that we can improve the weight spectrum by choosing appropriate PCT matrix to pre-transform the code, providing insights for designing codes with moderate blocklengths.

A. Minimum distance of pre-transformed monomial codes

It has already been shown in [9] that pre-transformation does not reduce minimum distance.

Lemma 4. *For any length-N monomial code $\mathcal{C}$ and any PT matrix $\boldsymbol{T} \in \mathcal{T}_N$, $d(\mathcal{C}) \le d(\mathcal{C}_T)$.*

Extending this result, however, is generally difficult since monomial codes is a rather large family. We now focus on a more restrictive family of codes, yet still inclusive enough to cover RM codes and polar codes.

Definition 2 (Monomial order, [12])**.** Two monomials of the same degree in $\mathcal{M}_n$ are ordered as $x_{i_1} \cdots x_{i_r} \preceq x_{j_1} \cdots x_{j_r}$ if and only if $i_s \le j_s$ for all $1 \le s \le r$, where we assume $i_1 < \cdots < i_r$ and $j_1 < \cdots < j_r$. For monomials $f, g \in \mathcal{M}_n$ of different degree, $f \preceq g$ if and only if there exists a divisor g^* of g of the same degree as f and $f \preceq g^*$.

Definition 3 (Decreasing monomial code, [12])**.** A monomial code $\mathcal{C}$ with information set $\mathcal{I}$ is said to be *decreasing* if $i \in \mathcal{I}$ and $f^{(j)} \preceq f^{(i)}$ together implies $j \in \mathcal{I}$.

Lemma 5 ([12]). *RM codes and polar codes are decreasing monomial codes.*

We now show that for a decreasing monomial code $\mathcal{C}$, pre-transformation does not change the minimum distance. Specially, our previous discussion allows us to limit our scope on PCT matrices rather than PT matrices.

Theorem 6. *For any length-N decreasing monomial code $\mathcal{C}$ and any PT matrix $\boldsymbol{S} \in \mathcal{S}_{\mathcal{C}}$, $d(\mathcal{C}) = d(\mathcal{C}_S)$.*

Proof. By Lemma 4, It suffices to prove that there exists a codeword in $\mathcal{C}_S$ with weight $d(\mathcal{C})$. Let $\mathcal{I}$ be the information set, then $N+1-2^r \in \mathcal{I}$, where $r = \log_2(N/d(\mathcal{C}))$. This is because

$$\mathrm{wt}(\boldsymbol{h}_N^{(N+1-2^r)}) = d(\mathcal{C}) \text{ and } f^{(N+1-2^r)} = \prod_{i=0}^{r-1} x_i \preceq f^{(i)}$$

for all $i \in \mathcal{I}$ and $\mathrm{wt}(\boldsymbol{h}^{(i)}) = \mathrm{wt}(\boldsymbol{h}^{(N+1-2^r)})$. Also note that $\forall\, j > N+1-2^r$, $f^{(j)} \mid f^{(N+1-2^r)}$, hence $j \in \mathcal{I}$. Consider any $\boldsymbol{S} \in \mathcal{S}_{\mathcal{C}}$, we claim that $\boldsymbol{s}^{(N+1-2^r)} = \boldsymbol{e}_{N+1-2^r}$, the $(N+1-2^r)$-th unit vector. The upper-triangularity of $\boldsymbol{S}$ requires that for all $1 \le j < N+1-2^r$, $S_{N+1-2^r,j} = 0$, and for $N+1-2^r < j \le N$, the second condition in Definition 1 requires $S_{N+1-2^r,j} = 0$ as $j \in \mathcal{I}$. Hence the following vector

$$\boldsymbol{e}_{N+1-2^r}\boldsymbol{S}\boldsymbol{H}_N = \boldsymbol{e}_{N+1-2^r}\boldsymbol{H}_N = \boldsymbol{h}_N^{(N+1-2^r)}$$

is a codeword of $\mathcal{C}_S$ which has weight $d(\mathcal{C})$. □

B. Number of minimum weight codewords of pre-transformed monomial codes

We have shown in the previous subsection that for decreasing monomial code $\mathcal{C}$, $d(\mathcal{C}_S) = d(\mathcal{C})$ for any $\boldsymbol{S} \in \mathcal{S}_{\mathcal{C}}$. We are now interested in the number of minimum weight codewords of a code $\mathcal{C}$:

$$M(\mathcal{C}) := |\{\boldsymbol{c} \in \mathcal{C} : \mathrm{wt}(\boldsymbol{c}) = d(\mathcal{C})\}|,$$

which can be seen as another 'metric' for the weight spectrum. For decreasing monomial code $\mathcal{C}$ satisfying certain assumptions, we aim to construct a PCT matrix $\boldsymbol{S} \in \mathcal{S}_{\mathcal{C}}$ such that $M(\mathcal{C}_S) < M(\mathcal{C})$. We now state the condition for $\mathcal{C}$.

Assumption 1. *For decreasing monomial code $\mathcal{C}$ with information set $\mathcal{I}$, let i^* be the smallest integer in $\mathcal{I}$ such that $\mathrm{wt}(\boldsymbol{h}^{(i^*)}) = d(\mathcal{C})$. Then $d(\mathcal{C}) < 2^{n-1}$ and there exists $\ell \in \mathcal{I}^C$ with $\ell > i^*$ s.t. $\mathrm{wt}(\boldsymbol{h}^{(\ell)}) < d(\mathcal{C})$.*

In addition, define the *sub-minimum distance* of a code $\mathcal{C}$:

$$d_{\mathrm{sub}}(\mathcal{C}) := \min\{\mathrm{wt}(\boldsymbol{c}') : \boldsymbol{c}' \in \mathcal{C}, \mathrm{wt}(\boldsymbol{c}') > d(\mathcal{C})\},$$

and let

$$e := \min\{e' \in \mathbb{N} : N - i^* > 2^{n-e'}\}$$

we have the following proposition:

Proposition 7. *For length-N decreasing monomial code $\mathcal{C}$ with information set $\mathcal{I}$ selected according to Assumption 1, suppose in addition that $d_{\mathrm{sub}}(\mathcal{C}) - d(\mathcal{C}) > 2^e$, there exists $\boldsymbol{S} \in \mathcal{S}_{\mathcal{C}}$ such that $M(\mathcal{C}_S) < M(\mathcal{C})$.*

Proof. For simplicity, let $r^* := \frac{N}{d(\mathcal{C})} - 1$ be the maximum degree of the monomials that generate the code. Define

$$L := N + 1 - 2^{n-e} > i^*.$$

Note that $f^{(L)} = \prod_{u=0}^{n-e-1} x_u$. It is not hard to verify that $L \in \mathcal{I}^C$: if $\deg(f^{(i^*)}) \ge n-e$ there must be $\ell > i^*$ s.t. $\deg(f^{(\ell)}) > n-e$, but this is impossible; if $\deg(f^{(i^*)}) < n-e = \deg(f^{(L)})$, $L \in \mathcal{I}^C$ by choice of i^*. Since $N+1-2^{n-e-1} < i^* \le N - 2^{n-e}$, $x_{n-e} \mid f^{(i^*)}$. We let K be the largest integer less than L s.t. $\deg(f^{(K)}) = r^*$. It is not hard to observe that

$$K = 2^{n-e} + 2^{r^*} - 1 \text{ and } f^{(K)} = x_{n-e} \cdot \textstyle\prod_{i=0}^{r^*-2} x_i$$

Clearly, by decreasing property, $K \in \mathcal{I}$.

Define $\boldsymbol{S}$ such that $S_{i,j} = 1$ iff $i = j$ or $(i,j) = (K,L)$. Obviously $\boldsymbol{S} \in \mathcal{S}_{\mathcal{C}}$. Note that for any bit sequence $\boldsymbol{u}$ such that $\mathrm{wt}(\boldsymbol{u}\boldsymbol{H}_N) > d(\mathcal{C})$,

$$\mathrm{wt}(\boldsymbol{u}\boldsymbol{S}\boldsymbol{H}_N) \ge \mathrm{wt}(\boldsymbol{u}\boldsymbol{H}_N) - \mathrm{wt}(\boldsymbol{h}_N^{(L)}) \ge d_{\mathrm{sub}}(\mathcal{C}) - 2^e > d(\mathcal{C}).$$

It suffices to consider the codewords in $\mathcal{C}$ that attain minimum weight. More explicitly, since pre-transformation does not reduce minimum distance, we only need to construct a minimum-weight codeword in $\mathcal{C}$ whose corresponding codeword in $\mathcal{C}_S$ has larger weight than $d(\mathcal{C})$. Define bit sequence $\boldsymbol{v}$ where $v_i = 1$ iff $i \in \{K, K+1\}$. The codeword $\boldsymbol{v}\boldsymbol{H}_N = \boldsymbol{h}_N^{(K)} \oplus \boldsymbol{h}_N^{(K+1)}$ is of minimum weight since

$$f^{(K+1)} = x_{n-e} \cdot \textstyle\prod_{i=1}^{r^*-2} x_i \;\Rightarrow f^{(K)} = x_0 \cdot f^{(K+1)},$$

indicating that $\boldsymbol{h}_N^{(K)}$ equals 1 at exactly half of the positions where $\boldsymbol{h}_N^{(K+1)}$ equals 1. Now $\boldsymbol{v}\boldsymbol{S}\boldsymbol{H}_N = \boldsymbol{v}\boldsymbol{H}_N \oplus \boldsymbol{h}_N^{(L)}$. Letting $\tilde{f} = f^{(K)} + f^{(K+1)}$, we directly have $(x_0+1) \mid \tilde{f}$, but meanwhile $x_0 \mid f^{(L)}$. These two polynomials do not evaluate to 1 simultaneously, therefore,

$$\mathrm{wt}(\boldsymbol{v}\boldsymbol{S}\boldsymbol{H}_N) = \mathrm{wt}(\boldsymbol{v}\boldsymbol{H}_N) + \mathrm{wt}(\boldsymbol{h}_N^{(L)}) > \mathrm{wt}(\boldsymbol{v}\boldsymbol{H}_N),$$

concluding the proof. □

We close this section by showing that for RM codes with particular rates, the previous proposition is applicable.

Corollary 8. *For RM codes $\mathcal{C} = \mathcal{R}(n,r)$ with $n-1 \ge 2r$, $r \ge 2$, there exists $\boldsymbol{S} \in \mathcal{S}_{\mathcal{C}}$ such that $M(\mathcal{C}_S) < M(\mathcal{C})$*

The proof is easily obtained by applying McEliece's Theorem [13, Corollary 13, Ch. 15].

V. Simulation

In this section, we verify the correctness of Algorithm 1. In particular, we transform PAC codes [8] to parity-check monomial codes, and then compare the weight spectrum of both codes.

We first briefly introduce PAC codes. A PAC code is specified by four parameters $(N, k, \mathcal{I}, \boldsymbol{g})$, where the (N,k)-code $\mathcal{C}$ is given by

$$\mathcal{C} = \{\boldsymbol{c} = \boldsymbol{u}\boldsymbol{T}\boldsymbol{H}_N : u_j = 0 \text{ if } j \notin \mathcal{I}\}.$$

The matrix $\boldsymbol{T}$ is an upper triangular Toeplitz matrix which serves as a convolution operation with length-$(m+1)$ impulse response $\mathbf{g} = (g_0, \cdots, g_m)$. By convention, $g_0 = g_m = 1$. It is hence straightforward that $\mathcal{C}$ falls into the category of pre-transformed monomial codes. It is commented in [8] that a heuristic method of choosing $\mathcal{I}$ is based on score function: $s : [N] \to \mathbb{R}$, and elements of $\mathcal{I}$ is selected to be the indices corresponding to the k-largest scores (with ties broken arbitrarily). One such score function is the Reed-Muller (RM) score function $s(i) = \mathrm{wt}((i-1)_2)$. Upon using this score function, we recover the RM codes when $\boldsymbol{T} = \boldsymbol{I}$.

In the following we consider two PAC codes $\mathcal{C}' = (32, 16, \mathcal{I}, \boldsymbol{g}')$, $\mathcal{C}'' = (32, 16, \mathcal{I}, \boldsymbol{g}'')$ with $\mathcal{I}$ selected according to the RM score function, and $\boldsymbol{g}' = (1, 1, 0, 1)$, $\boldsymbol{g}'' = (1, 0, 1, 1, 0, 1)$. In addition, we let $\mathcal{C}$ be the corresponding Reed-Muller code without pre-transformation, or alternatively, $\mathcal{C} = (32, 16, \mathcal{I}, (1))$. We list our results below.

TABLE I
WEIGHT SPECTRUM OF $\mathcal{C} = (32, 16, \mathcal{I}, (1))$

0	8	12	16	20	24	32
1	620	13888	36518	13888	620	1

TABLE II
SIMULATION RESULTS OF $\mathcal{C}' = (32, 16, \mathcal{I}, (1, 1, 0, 1))$

Weight Spectrum of $\mathcal{C}' = (32, 16, \mathcal{I}, (1, 1, 0, 1))$										
0	8	10	12	14	16	18	20	22	24	32
1	364	2048	6720	14336	18598	14336	6720	2048	364	1
Weight Spectrum of Transformed PC Monomial Code										
0	8	10	12	14	16	18	20	22	24	32
1	364	2048	6720	14336	18598	14336	6720	2048	364	1
Parity Check Equations										
$u_9 \oplus u_8 = 0$ $u_{11} \oplus u_8 = 0$ $u_{13} \oplus u_{12} = 0$ $u_{17} \oplus u_{12} \oplus u_{15} \oplus u_{16} = 0$ $u_{18} \oplus u_{12} \oplus u_{14} \oplus u_{15} = 0$ $u_{19} \oplus u_{12} \oplus u_{14} \oplus u_{15} \oplus u_{16} = 0$ $u_{21} \oplus u_{20} = 0$ $u_{25} \oplus u_{20} \oplus u_{23} \oplus u_{24} = 0$										

TABLE III
SIMULATION RESULTS OF $\mathcal{C}'' = (32, 16, \mathcal{I}, (1, 0, 1, 1, 0, 1))$

Weight Spectrum of $\mathcal{C}'' = (32, 16, \mathcal{I}, (1, 0, 1, 1, 0, 1))$										
0	8	10	12	14	16	18	20	22	24	32
1	492	1024	10304	7168	27558	7168	10304	1024	492	1
Weight Spectrum of Transformed PC Monomial Code										
0	8	10	12	14	16	18	20	22	24	32
1	492	1024	10304	7168	27558	7168	10304	1024	492	1
Parity Check Equations										
$u_{10} \oplus u_8 = 0$ $u_{11} \oplus u_8 = 0$ $u_{13} \oplus u_8 = 0$ $u_{17} \oplus u_{12} \oplus u_{14} \oplus u_{15} = 0$ $u_{18} \oplus u_{14} \oplus u_{15} \oplus u_{16} = 0$ $u_{19} \oplus u_{16} = 0$ $u_{21} \oplus u_{12} \oplus u_{14} \oplus u_{16} = 0$ $u_{25} \oplus u_{12} \oplus u_{15} \oplus u_{20} \oplus u_{22} \oplus u_{23} = 0$										

In both examples we observe that the weight spectra of the PAC code and its corresponding PC monomial code are the same, which supports the validity of our algorithm. Furthermore, the numbers of minimum weight codewords in both examples are strictly less than that of the original RM code. This is in accordance to the comment in [8], i.e., choosing $\boldsymbol{g}$ at random might be an acceptable design.

VI. CONCLUSION

In this paper, we give a theorem and an algorithm to establish the equivalence between pre-transformed monomial codes and parity-check monomial codes. This result emphasizes the theoretical importance of PC monomial codes proposed in [7], in the sense that practical codes such as PAC codes belong to the PC monomial code family. Furthermore, for certain decreasing monomial codes, specific pre-transformation matrices can be constructed to strictly improve the weight spectrum. Combining these two results, we know that pre-transformation can indeed improve the code performance of polar or Reed-Muller codes and the searching space can be reduced by only designing the parity-check bits. However, how to design these parity-check bits in order to obtain optimal (or near-optimal) improvement of the weight spectrum is still an open problem.

REFERENCES

[1] E. Arikan, "Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels," *IEEE Transactions on information Theory*, vol. 55, no. 7, pp. 3051–3073, 2009.

[2] I. Tal and A. Vardy, "List decoding of polar codes," *IEEE Transactions on Information Theory*, vol. 61, no. 5, pp. 2213–2226, 2015.

[3] P. Trifonov and V. Miloslavskaya, "Polar subcodes," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 2, pp. 254–266, 2015.

[4] P. Yuan, T. Prinz, G. Böcherer, O. Iscan, R. Boehnke, and W. Xu, "Polar code construction for list decoding," in *SCC 2019; 12th International ITG Conference on Systems, Communications and Coding*. VDE, 2019, pp. 1–6.

[5] K. Niu and K. Chen, "Crc-aided decoding of polar codes," *IEEE communications letters*, vol. 16, no. 10, pp. 1668–1671, 2012.

[6] B. Li, H. Shen, and D. Tse, "A rm-polar codes," *arXiv preprint arXiv:1407.5483*, 2014.

[7] H. Zhang, R. Li, J. Wang, S. Dai, G. Zhang, Y. Chen, H. Luo, and J. Wang, "Parity-check polar coding for 5g and beyond," in *2018 IEEE International Conference on Communications (ICC)*. IEEE, 2018, pp. 1–7.

[8] E. Arıkan, "From sequential decoding to channel polarization and back again," *arXiv preprint arXiv:1908.09594*, 2019.

[9] B. Li, H. Zhang, and J. Gu, "On pre-transformed polar codes," *arXiv preprint arXiv:1912.06359*, 2019.

[10] Y. Li, H. Zhang, R. Li, J. Wang, G. Yan, and Z. Ma, "On the weight spectrum of pre-transformed polar codes," *arXiv preprint arXiv:2102.12625*, 2021.

[11] E. Abbe, A. Shpilka, and M. Ye, "Reed–muller codes: Theory and algorithms," *IEEE Transactions on Information Theory*, vol. 67, no. 6, pp. 3251–3277, 2020.

[12] M. Bardet, V. Dragoi, A. Otmani, and J.-P. Tillich, "Algebraic properties of polar codes from a new polynomial formalism," in *2016 IEEE International Symposium on Information Theory (ISIT)*. IEEE, 2016, pp. 230–234.

[13] F. J. MacWilliams and N. J. A. Sloane, *The theory of error correcting codes*. Elsevier, 1977, vol. 16.