Generalized and Extended Subspace Algorithms for Error Correction with Quantized DFT Codes

Mojtaba Vaezi, *Student Member, IEEE*, and Fabrice Labeau, *Senior Member, IEEE*

***Abstract*—Discrete Fourier transform (DFT) codes have been used to provide robustness against errors and erasures in various applications. This paper focuses on improving error localization of the *Bose-Chaudhuri-Hocquenghem* (BCH) DFT codes. First, we analyze how the *subspace-based error localization* outperforms the coding-theoretic one. Then, we propose an extension of the subspace-based error localization, based on additional syndrome, that improves the existing one and is naturally suitable for *rate-adaptive* distributed source coding (DSC). Further, we propose a new generic subspace-based algorithm to decode BCH-DFT codes. The proposed approach generalizes the encoding and decoding of this important class of DFT codes. It introduces many different decoding matrices for a DFT code; this *diversity* is then used to diminish the effect of the quantization noise and thus to improve the decoding. Finally, the *extended* and *generalized* approaches are combined to maximize the decoding gain. Simulation results demonstrate the capability of the proposed algorithms to perform significantly better than the existing subspace-based error localization, in the presence of quantization noise.**

***Index Terms*—BCH-DFT codes, subspace error localization, distributed source coding, channel coding, rate-adaptive codes.**

I. Introduction

THE ideas of *coding theory* can be described within the *signal processing* realm by virtue of a class of complex or real BCH codes [1], [2, Chapter 6], known as the *discrete Fourier transform codes*. DFT codes find applications a wide range of areas including wireless communications [3], [4], joint source-channel coding [5], distributed source coding [6], and compressive sensing [7]. Looking from the *frame theory* perspective, these codes are used to provide robustness against packet erasure in wireless networks [8]–[10].

Compared with the finite-field channel codes, DFT codes are preferred for their unconstrained codeword length, fast and easy implementation with floating-point operations, and ability in alleviating the quantization error besides correcting the errors (erasures) introduced by the transmission channel. On the other hand, there is a fundamental difference between decoding of binary BCH codes and that of BCH-DFT codes. To be transmitted over a digital communication channel, the codewords of any BCH-DFT code have to be quantized; this adds noise to each sample of every codeword, in channel coding. Similarly, in distributed source coding based on BCH-DFT codes, the parity or syndrome samples are to be quantized before transmission [6]. In any of those cases, the syndrome samples, formed at the decoder, are affected by quantization noise; thereby, the *syndrome decoding* of errors is feasible only if the *quantization error* is very small compared to the impulse noise.

In all applications where error correction is required [3]–[7], *error localization* is a crucial step of the decoding algorithm in DFT codes. For a BCH-DFT code, error localization can be carried out by extending the coding-theoretic syndrome decoding of binary BCH codes to the real field [2, Chapter 6]. By adopting the MUSIC- [11] and ESPRIT-based [12] methods of multiple frequency component estimation, Rath and Guillemot [13] proposed *subspace-based* error localization algorithms which perform better than the *coding-theoretic* one. The above subspace-based approaches have been successfully applied to other real-number codes, such as the *discrete cosine transform* (DCT) and *discrete sine transform* (DST) codes [14], [15], which are based on orthogonal transform matrices. Between the two subspace-based methods, the MUSIC-like approach performs slightly better than the ESPRIT-like approach; it is also the most accurate error localization method for quantized DFT codes, to date.

Our main contribution in this paper is to further improve the error localization of quantized DFT codes. To put our results in perspective, we first analyze why the subspace-based error localization outperforms the coding-theoretic one. The key is that the subspace-based method is capable of providing more than one error locating polynomial. These polynomials have the same set of roots if the code is not quantize; however, they may result in different roots for quantized codes. Even so, by averaging the coefficients of polynomials, one can diminish the effect of the quantization error. With this insight, and based on additional syndrome samples, we extend the subspace error localization to further increase the number of polynomials and improve error localization. The proposed algorithm is naturally suitable for rate-adaptive distributed source coding; it is also applicable to channel coding in specific cases. It can be used with DCT and DST codes, too.

Another major contribution towards improving the subspace error localization is achieved by generalizing this algorithm through constructing various decoding matrices. We first prove that for an (n, k) DFT code there are $\phi(n)$ syndrome matrices for decoding, where $\phi(n)$ is the number of positive integers less than n that are *relatively prime* to n. We then exploit this

Manuscript received June 14, 2013; revised September 27, 2013. The editor coordinating the review of this paper and approving it for publication was K. Abdel-Ghaffar.

This work was supported by Hydro-Québec, the Natural Sciences and Engineering Research Council of Canada and McGill University in the framework of the NSERC/Hydro-Québec/McGill Industrial Research Chair in Interactive Information Infrastructure for the Power Grid.

Part of the material in this paper was presented at the International Symposium on Information Theory, Istanbul, Turkey, July 2013.

The authors are with the Department of Electrical and Computer Engineering, McGill University, Montreal, QC H3A 0E9, Canada (e-mail: mojtaba.vaezi@mail.mcgill.ca; fabrice.labeau@mcgill.ca).

Digital Object Identifier 10.1109/TCOMM.2014.010414.130440

diversity to improve the error localization by combining the error locating polynomials corresponding to different matrices. This algorithm is referred to as the *generalized subspace* method; it is combined with the *extended subspace* method to further improve the decoding.

Apart from the diversity in decoding, the generalized subspace method brings another novelty at the encoder side; that is, the parity frequencies of BCH-DFT codes (equivalently, the zeros of codewords in the frequency domain) are not required to be cyclically adjacent. This provides substantial flexibility in constructing BCH-DFT codes. Finally, in addition to improving the error localization, both extended and generalized subspace algorithms can be applied to increase the accuracy of the error detection step.

The paper is organized as follows. After a brief review of DFT codes in Section II, we investigate how the MUSIC-like subspace error localization beats the coding-theoretic one, in Section III. This facilitates introducing the extended subspace decoding in Section IV. We present the generalized subspace decoding in Section V. These two algorithms are combined in Section VI and their applications are illustrated in Section VII by means of a few examples. Numerical results in Section VIII confirm the merit of the proposed error localization algorithms. This is followed by conclusions in Section IX.

II. DFT Codes

We define DFT codes and introduce their basic properties in this section.

Definition 1. An $n \times n$ DFT matrix is defined by

$$W_n \triangleq \frac{1}{\sqrt{n}} \begin{pmatrix} 1 & 1 & 1 & \cdots & 1 \\ \omega^0 & \omega^1 & \omega^2 & \cdots & \omega^{n-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \omega^0 & \omega^{n-1} & \omega^{2(n-1)} & \cdots & \omega^{(n-1)(n-1)} \end{pmatrix}, \quad (1)$$

in which $\omega = e^{-j\frac{2\pi}{n}}$ [16]. Let W_n^H denote *conjugate transpose* of the DFT matrix. Since W_n is unitary, $W_n^H = W_n^{-1}$; i.e., W_n^H gives the inverse DFT (IDFT) matrix.

The generator matrix of an (n, k) DFT code [1], in general, consists of any k columns of the IDFT matrix of order n; the remaining $n - k$ columns of this matrix are used to build the *parity-check matrix*, as we will see shortly in this section. These codes are a family of *cyclic codes* over the complex field. Thus, their codewords satisfy certain spectral properties in the frequency domain [17, Chapter 4]. Within the class of DFT codes, there are BCH codes in the *complex* and *real* fields. The generator matrix of an (n, k) real BCH-DFT code is commonly defined by [5], [18]

$$G = \sqrt{\frac{n}{k}} W_n^H \Sigma W_k, \quad (2)$$

where W_n and W_k are the DFT matrices of size n and k, and

$$\Sigma = \begin{pmatrix} I_\alpha & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \\ \mathbf{0} & I_\beta \end{pmatrix} \quad (3)$$

is an $n \times k$ matrix with $\alpha = \lceil \frac{n}{2} \rceil - \lfloor \frac{n-k}{2} \rfloor$ and $\alpha + \beta = k$. I_α represents the identity matrix of size α and alpha is the frequency offset required to obtain a real matrix G. In other words, it identifies the indices in which zeros must be inserted in order to get the conjugacy constraint required to have a real-valued G [17, Fig. 4.2], [19]. A complex BCH-DFT code is obtained by removing W_k from (2). We can also remove the first constraint on α; that is, for a complex code, α can be any integer between 1 and k.

The *parity-check matrix* H, both for real and complex BCH-DFT codes, is then defined as

$$H = \begin{pmatrix} 1 & 1 & \cdots & 1 \\ \omega^{\alpha} & \omega^{\alpha+1} & \cdots & \omega^{n-\beta-1} \\ \vdots & \vdots & \ddots & \vdots \\ \omega^{\alpha(n-1)} & \omega^{(\alpha+1)(n-1)} & \cdots & \omega^{(n-\beta-1)(n-1)} \end{pmatrix}^H. \quad (4)$$

Obviously, H is an $(n - k) \times n$ matrix and

$$HG = \mathbf{0}_{n-k \times k} \quad (5)$$

both for real and complex codes. Each codeword of an (n, k) BCH-DFT code has

$$d \triangleq n - k \quad (6)$$

cyclically adjacent zeros in the frequency domain. These codes are *maximum distance separable* (MDS) codes with *minimum Hamming distance* $d_{\min} = d + 1$. They are, hence, capable of correcting up to $t = \lfloor \frac{d}{2} \rfloor$ errors.

In general, every sample in the codewords of a DFT code is a linear combination of all data samples of the input block, i.e., the data samples do not appear explicitly in the codewords. A specific method of encoding, known as *systematic encoding*, leaves the data samples unchanged. These unchanged samples can be exhibited in any k components of the codeword, but evenly-spaced data samples minimizes the variance of parity samples, for a given input variance [18, Section VII]. Hence, such a code reduces required transmission rate for a fixed distortion, in view of rate distortion theory [20, Chapter 10]. Systematic codes are particularly important in the context of distributed source coding [6], [21].

In the rest of this paper, for brevity, BCH-DFT codes will be referred to as DFT codes.

III. Error Localization in DFT codes: A Review

Let the $n \times 1$ vector $\boldsymbol{c}$ represent a codeword generated by an (n, k) DFT code. Also, let $\boldsymbol{r} = \boldsymbol{c} + \boldsymbol{e}$ be a noisy version of $\boldsymbol{c}$ and suppose that the error vector $\boldsymbol{e}$ has $\nu \leq t$ nonzero elements. Let $1 \leq i_1, \ldots, i_\nu \leq n$ and $e_{i_1}, \ldots, e_{i_\nu}$ denote, respectively, the locations and magnitudes of the nonzero elements of $\boldsymbol{e}$. The decoding algorithm in DFT codes is composed of three main steps [2]: *error detection* (to determine ν), *error localization* (to find $i_1, \ldots, i_\nu$), and *error calculation* (to calculate $e_{i_1}, \ldots, e_{i_\nu}$). This paper is mainly focused on improving the error localization step. Thus, for the moment during the development, we assume that the number of errors (i.e., ν) is known at the decoder. Later, it will be apparent how to determine the true value of ν at the decoder. This is part of the spectral decomposition and will be briefly discussed in Section III-B.

The syndrome of $\boldsymbol{e}$, which is the key input of the decoding algorithm, is computed as

$$\boldsymbol{s} = H\boldsymbol{r} = H(\boldsymbol{c}+\boldsymbol{e}) = H\boldsymbol{e}, \tag{7}$$

where $\boldsymbol{s} = [s_1, s_2, \ldots, s_d]^T$ is a complex vector with

$$s_\ell = \frac{1}{\sqrt{n}} \sum_{p=1}^{\nu} e_{i_p} X_p^{\alpha-1+\ell}, \quad \ell = 1, \ldots, d, \tag{8}$$

in which α is defined in (3) and

$$X_p = \omega^{-i_p}, \qquad p = 1, \ldots, \nu. \tag{9}$$

where $\omega = e^{-j\frac{2\pi}{n}}$. It is worth noting that $X_1, \ldots, X_n$ are the nth roots of unity and $X_1, \ldots, X_\nu$ indicate those roots which correspond to the error indices, as we have assumed $i_1, \ldots, i_\nu$ to be the error indices. There are two main approaches to find the error indices; we describe them in the following.

A. Coding-Theoretic Approach

The classical approach to the error localization is to identify an *error-locator polynomial* whose roots correspond to error locations. The error-locator polynomial can be defined as

$$\Lambda(x) = \prod_{i=1}^{\nu}(1 - xX_i^{-1}) = 1 + \Lambda_1 x + \cdots + \Lambda_\nu x^\nu, \tag{10}$$

and its roots $X_1, \ldots, X_\nu$ correspond to the error locations i_p, $p \in [1, \ldots, \nu]$, as from (9) we have $i_p = \frac{\log X_p}{\log \omega^{-1}} = \frac{\arg X_p}{\arg \omega^{-1}}$. Note that for a complex z we have $\log z = \ln|z| + j \arg z$ and we take $\arg z \in (0, 2\pi]$. The coefficients $\Lambda_1, \ldots, \Lambda_\nu$ can be found by solving the following set of *consistent* equations [2]

$$s_j\Lambda_\nu + s_{j+1}\Lambda_{\nu-1} + \cdots + s_{j+\nu-1}\Lambda_1 = -s_{j+\nu}, \tag{11}$$

for $j = 1, \ldots, d-\nu$. To put it differently, as the IDFT of $\boldsymbol{\Lambda}_n = [1, \Lambda_1, \ldots, \Lambda_\nu, \boldsymbol{0}_{1\times(n-\nu-1)}]^T$ becomes zero at the error locations, the circular convolution of $\boldsymbol{\Lambda}_n$ with the DFT of the error vector is a zero vector [2], [13].

B. Subspace-Based Approach: A Fresh Look

Alternatively, one can use a *subspace-based* method for error localization [13]. We elaborate the subspace-based method which is along the lines of the MUSIC [11], since it is shown to perform better than the ESPRIT-like [12] method, in [13]. To this end, let $\nu+1 \le m \le d-\nu+1$ and define the following syndrome matrix

$$S_m = \begin{bmatrix} s_1 & s_2 & \cdots & s_{d-m+1} \\ s_2 & s_3 & \cdots & s_{d-m+2} \\ \vdots & \vdots & \ddots & \vdots \\ s_m & s_{m+1} & \cdots & s_d \end{bmatrix}, \tag{12}$$

whose elements are given by (7). Also, define the covariance matrix as

$$R_m = S_m S_m^H. \tag{13}$$

Subspace-based error localization is based on eigendecomposition of R_m. Before proceeding to the details of the algorithm, it is important to point out that S_m can be decomposed as

$$S_m = V_m D V_{d-m+1}^T, \tag{14}$$

in which V_m is a *Vandermonde* matrix defined as

$$V_m = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ X_1 & X_2 & \cdots & X_\nu \\ \vdots & \vdots & \ddots & \vdots \\ X_1^{m-1} & X_2^{m-1} & \cdots & X_\nu^{m-1} \end{bmatrix}, \tag{15}$$

and D is a diagonal matrix of size ν with nonzero diagonal elements $d_p = \frac{1}{\sqrt{n}} e_{i_p} X_p^\alpha, p = 1, \ldots, \nu$. V_m is called the *error-locator matrix* of order m [13], and its columns are the *error-locator vectors* of order m. Since $m > \nu$, the columns of V_m define a ν-dimensional subspace of the m-dimensional vector space, which is referred to as the *channel-error subspace*. Its orthogonal complement subspace is called the *noise subspace* and has dimension $m - \nu$.

One can verify that the rank of S_m is ν. To check this, from linear algebra, we know that $\mathrm{rank}(AB) = \mathrm{rank}(A)$ if B has a full column rank [22]. Then, by twice applying this to (14) and recalling that $\nu + 1 \le m \le d - \nu + 1$ its is easy to see that $\mathrm{rank}(S_m) = \mathrm{rank}(V_m) = \nu$. From this, it is obvious that the rank of R_m is ν;[1] thus, it can be eigendecomposed as

$$R_m = [U_\mathrm{e}\ U_\mathrm{q}] \begin{bmatrix} \Delta_\mathrm{e} & \boldsymbol{0} \\ \boldsymbol{0} & \Delta_\mathrm{q} \end{bmatrix} [U_\mathrm{e}\ U_\mathrm{q}]^H, \tag{16}$$

where the square matrices Δ_e and Δ_q contain the ν largest and $m - \nu$ smallest eigenvalues, and U_e and U_q contain the eigenvectors corresponding to Δ_e and Δ_q, respectively.[2] The sizes of U_e and U_q are $m \times \nu$ and $m \times (m-\nu)$. The columns in U_e span the channel-error subspace [13, Proposition 1] spanned by V_m. Thus, from the fact that $U_\mathrm{e}^H U_\mathrm{q} = \boldsymbol{0}$, we conclude that

$$V_m^H U_\mathrm{q} = \boldsymbol{0}. \tag{17}$$

Now, let $\boldsymbol{v} = [1, x, x^2, \ldots, x^{m-1}]^T$ where x is a variable that can take on any of $X_1, \ldots, X_n$. We define the function

$$F(x) \triangleq \sum_{j=1}^{m-\nu} \boldsymbol{v}^H \boldsymbol{u}_{q,j} = \sum_{j=1}^{m-\nu} \sum_{k=0}^{m-1} f_{jk} x^k, \tag{18}$$

where $\boldsymbol{u}_{q,j}$ represents the jth column of U_q. $F(x)$ can be considered as sum of $m - \nu$ polynomials $\{f_j\}_{j=1}^{m-\nu}$ of order $m-1$; each polynomial is derived from a column of U_q. Let $\mathcal{F}$ denote this set of polynomials restricted to coefficients from U_q. In light of (17), each one of these polynomials vanishes for $x = X_1, \ldots, X_\nu$, i.e., $F(x) = 0$ for $X_1, \ldots, X_\nu$. These are the only common roots of $\{f_j\}$ over the nth roots of unity [13];[3] thus, the errors location can be determined by finding the zeros of $F(x)$ over the set of nth roots of unity. It should be mentioned that the above algorithm is based on the *noise subspace*. One may, equivalently, use the *signal subspace* to find the error locations [23]. Also, the above approach is along

[1] From the above argument one can see that the number of errors can be found by evaluating the rank of S_m or R_m for any $\nu+1 \le m \le d-\nu+1$. Thus, since we do not know ν, it is better to choose $m = \lfloor \frac{d}{2} \rfloor$ or $m = \lceil \frac{d}{2} \rceil$, to detect as many errors as possible.

[2] Clearly, since no noise (or quantization error) is considered at this stage, $\Delta_\mathrm{q} = \boldsymbol{0}$ and Δ_e contains the ν nonzero eigenvalues of R_m.

[3] Assume, for the sake of contradiction, that there are $\nu+1$ common roots of unity. This implies that V_m has another column (corresponding to $X_{\nu+1}$) for which (17) holds; i.e., there are more than ν errors which is contradicting.

the lines of the MUSIC [11]; it is shown to perform slightly better than the ESPRIT-like [12] method, in [13].

The subspace method outperforms the coding-theoretic error localization. To see this, we can see that $\Lambda(x)$ is the smallest degree polynomial that has roots in $X_1, \ldots, X_\nu$ and lies in the noise subspace; it is achieved for $m = \nu+1$ in (18). As m increases, the degree of polynomials $\{f_j\}$ goes up which gives more *degrees of freedom* (DoF) and helps improve the estimation of the roots, and the error locations consequently. An even more important factor that affects location estimation is the number of polynomials $\{f_j\}$ whose coefficients come from *linearly independent* columns of $U_{\rm q}$. The more there are such polynomials, the better the estimation can be as the variation due to noise (quantization) is reduced by adding such independent polynomials in $F(x)$.

Although the number of polynomials increases with m, their coefficients may not be independent. The latter depends on the number of nonzero eigenvalues in the noise subspace which is, in turn, related to the rank of S_m and is limited by

$$\operatorname{rank}(S_m) \le \max_m \min(m, d-m+1) = \left\lceil \frac{d}{2} \right\rceil. \tag{19}$$

This suggests that the optimum value for m is $\lceil \frac{d}{2} \rceil$. Then, based on (18), one can expect that, in the presence of noise, the subspace approach will result in an error localization better than the coding-theoretic approach, except when $\nu = t$ and d is even; this is confirmed by simulation results in [13]. In the last case where $\nu = t$ we have $m = \nu + 1$ and there is just one polynomial and its degree is ν, the same as (10) in the coding-theoretic approach. In general, $F(x)$ is composed of

$$\mathcal{N}(m) = \min(m, d-m+1) - \nu \le \left\lceil \frac{d}{2} \right\rceil - \nu \tag{20}$$

independent polynomials where the upper bound is obtained for $m = \lceil \frac{d}{2} \rceil$. We will use $\mathcal{N}(m)$ as a measure of the gain introduced by the subspace-based error localization compared with the coding-theoretic approach.

C. Quantization Effect

In practice, where quantization comes into play, the received vector is distorted both by the error vector $\boldsymbol{e}$ and quantization noise $\boldsymbol{q}$. Therefore $\boldsymbol{r} = \boldsymbol{c} + \boldsymbol{e} + \boldsymbol{q}$, and its syndrome is only a perturbed version of $\boldsymbol{s}$ because

$$H\boldsymbol{r} = H(\boldsymbol{c} + \boldsymbol{q} + \boldsymbol{e}) = \boldsymbol{s}_{\rm q} + \boldsymbol{s} = \tilde{\boldsymbol{s}}, \tag{21}$$

where $\boldsymbol{s}_{\rm q} \equiv H\boldsymbol{q}$ and $\boldsymbol{q} = [q_1, q_2, \ldots, q_n]^T$ is the quantization error. The distorted syndrome samples can be written as

$$\begin{aligned} \tilde{s}_\ell &= \frac{1}{\sqrt{n}} \sum_{p=1}^{\nu} e_{i_p} X_p^{\alpha-1+\ell} + \frac{1}{\sqrt{n}} \sum_{p'=1}^{n} q_{i_{p'}} X_{p'}^{\alpha-1+\ell} \\ &= s_\ell + \frac{1}{\sqrt{n}} \sum_{r=1}^{n} q_r \omega^{(\alpha-1+\ell)r}, \quad 1 \le \ell \le d, \end{aligned} \tag{22}$$

where $i_{p'}$ shows the index for quantization error. The distorted syndrome matrix $\tilde{S}_m$ and its corresponding covariance matrix $\tilde{R} = \tilde{S}_m \tilde{S}_m^H$ are defined similar to (12) and (13) but for the distorted syndrome samples. Then, we can write

$$\tilde{R}_m = [\tilde{U}_{\rm e}\ \tilde{U}_{\rm q}] \begin{bmatrix} \tilde{\Delta}_e & \mathbf{0} \\ \mathbf{0} & \tilde{\Delta}_{\rm q} \end{bmatrix} [\tilde{U}_{\rm e}\ \tilde{U}_{\rm q}]^H, \tag{23}$$

where $\tilde{U}_{\rm e}$ and $\tilde{U}_{\rm q}$ span the "estimated" channel error and quantization noise subspaces, respectively. Due to the quantization error, these estimated subspaces are perturbations of the channel and noise subspaces, that is $\tilde{U}_{\rm e} = U_{\rm e} + U_{\rm qe}$ and $\tilde{U}_{\rm q} = U_{\rm q} + U_{\rm qq}$. Consequently, unlike (17), $V_m^H \tilde{U}_{\rm q} = V_m^H U_{\rm qq} \triangleq \Delta_{\tilde{F}} \neq \mathbf{0}$, or equivalently, the resulting polynomial

$$\tilde{F}(x) \triangleq \sum_{j=1}^{m-\nu} \boldsymbol{v}^H \tilde{\boldsymbol{u}}_{q,j} = \sum_{j=1}^{m-\nu} \sum_{k=0}^{m-1} \tilde{f}_{jk} x^k, \tag{24}$$

does not necessarily have roots at the nth roots of unity. Hence, among those n roots, ν roots that result in the smallest $|\tilde{F}(x)|$ are used to estimate error locations.

IV. Extended Subspace Approach

Error localization is a crucial step in the decoding algorithm of DFT codes. Numerical results show [19, Fig. 2 and 3] that if the location of errors are known at the decoder, reconstruction error can be less than quantization error. It is known that an (n, k) DFT code decreases the mean-squared reconstruction error (MSE) by a factor of code rate $r = \frac{k}{n}$ [8], [9], [19]. This motivates the search for methods that can further improve the error localization in DFT codes. In particular, we are interested to know whether error localization can be improved without or with extra syndrome samples. Since our objective is to improve the error localization performance, we assume that the number of errors ν is known at the decoder.

The main idea behind the *extended subspace* approach is to try to enlarge the dimension of the estimated noise subspace such that the number of polynomials with linearly independent coefficients and/or their degree grow, in (18). We observe that if we are able to construct a syndrome matrix S'_m such that it can be decomposed as

$$S'_m = V_m D V_{d'-m+1}^T, \tag{25}$$

for $d' > d$ and $\nu + 1 \le m \le d' - \nu + 1$, where V_m and D are the same as those in (14), then we can expect a better estimation for the location of errors. This is because following the same argument that led to (18) and (19) it is easy to see that, if S'_m is used for error localization, the optimal m in this case is $\lceil \frac{d'}{2} \rceil$ which results in $\lceil \frac{d'}{2} \rceil - \nu \ge \lceil \frac{d}{2} \rceil - \nu$ error-locator polynomials. Then, as explained in Section III-B, this can improve the error localization.

The challenge is to find the entries of the extended syndrome matrix S'_m. That is, we need to find s'_ℓ for all $1 \le \ell \le d'$ so as to build

$$S'_m = \begin{bmatrix} s'_1 & s'_2 & \cdots & s'_{d'-m+1} \\ s'_2 & s'_3 & \cdots & s'_{d'-m+2} \\ \vdots & \vdots & \ddots & \vdots \\ s'_m & s'_{m+1} & \cdots & s'_{d'} \end{bmatrix}. \tag{26}$$

From (25), it can be verified that

$$s'_\ell = \frac{1}{\sqrt{n}} \sum_{p=1}^{\nu} e_{i_p} X_p^{\alpha-1+\ell}, \quad \ell = 1, \ldots, d'. \tag{27}$$

Comparing with (8), it is clear that $s'_\ell = s_\ell$ for $1 \le \ell \le d$. Thus, we only need to determine the entries for $d < \ell \le d'$.

With this in mind, similar to the syndrome vector $\boldsymbol{s}$, we can define the *extended syndrome vector* $\bar{\boldsymbol{s}}$ as

$$\bar{\boldsymbol{s}} = \bar{H}\boldsymbol{e}, \tag{28}$$

where $\bar{H}$ consists of those k columns of the IDFT matrix of order n used to build G. In other words, $\bar{H}$ is the complement of H. More precisely, similar to (4),

$$\bar{H} \triangleq \begin{pmatrix} 1 & \cdots & 1 \\ \omega^{n-\beta} & \cdots & \omega^{n+\alpha-1} \\ \vdots & \ddots & \vdots \\ \omega^{(n-\beta)(n-1)} & \cdots & \omega^{(n+\alpha-1)(n-1)} \end{pmatrix}^H. \tag{29}$$

Note that $\bar{H}$ is a $k \times n$ matrix and, from (28), for $\ell = 1, \ldots, k$, we have

$$\bar{s}_\ell = \frac{1}{\sqrt{n}} \sum_{p=1}^{\nu} e_{i_p} X_p^{d+\alpha-1+\ell}. \tag{30}$$

Now, we can see that

$$s'_\ell = \begin{cases} s_\ell, & 1 \le \ell \le d, \\ \bar{s}_{\ell-d}, & d < \ell \le d', \end{cases} \tag{31}$$

where $d' - d$, $d' - d \le k$, is the number of extra syndromes.

So far we have shown that if we are able to compute (28) then we can form the extended syndrome matrix in (26), and thereby benefit from the larger number of polynomials it gives, when compared to (12). But how can we compute $\bar{\boldsymbol{s}}$ at the receiver? Obviously, we do not know $\boldsymbol{e}$ at the decoder; instead, we have $\boldsymbol{r} = \boldsymbol{c} + \boldsymbol{e}$. Seeing that

$$\bar{H}\boldsymbol{r} = \bar{H}\boldsymbol{e} + \bar{H}\boldsymbol{c} = \bar{\boldsymbol{s}} + \bar{H}\boldsymbol{c} \tag{32}$$

we will get $\bar{\boldsymbol{s}}$ provided that $\bar{H}\boldsymbol{c}$, the second term on the right-hand side of (32), is either removed or becomes zero. Observe that despite the fact that $H\boldsymbol{c} = \boldsymbol{0}$, $\bar{H}\boldsymbol{c}$ is not necessarily zero. In Section VII, we accomplish this for source coding with side-information available at the decoder, or more generally for DSC. To this end, for the source $\boldsymbol{x}$, the encoder computes and transmits $\bar{\boldsymbol{s}}_x = \bar{H}\boldsymbol{x}$ to the decoder. At the decoder, we have access to side information $\boldsymbol{y} = \boldsymbol{x} + \boldsymbol{e}$ and we can compute $\bar{\boldsymbol{s}}_y = \bar{H}\boldsymbol{y} = \bar{\boldsymbol{s}}_x + \bar{\boldsymbol{s}}_e$. From this $\bar{\boldsymbol{s}} = \bar{\boldsymbol{s}}_e = \bar{\boldsymbol{s}}_y - \bar{\boldsymbol{s}}_x$. Note that here $\boldsymbol{x}$ plays the role of $\boldsymbol{c}$.

Finally, considering quantization, $\boldsymbol{c}$ will be replaced by $\boldsymbol{c} + \boldsymbol{q}$; i.e., the new syndrome $\tilde{\bar{s}}_\ell$ will contain a term related to quantization error, similar to $\tilde{s}_\ell$ in (22). Likewise, $\tilde{s}'_\ell$ is built upon $\tilde{s}_\ell$ and $\tilde{\bar{s}}_\ell$, that is

$$\tilde{s}'_\ell = \begin{cases} \tilde{s}_\ell, & 1 \le \ell \le d, \\ \tilde{\bar{s}}_{\ell-d}, & d < \ell \le d', \end{cases} \tag{33}$$

where $\tilde{\boldsymbol{s}} = \boldsymbol{s}_e + \boldsymbol{s}_{\mathrm{q}}$, $\tilde{\bar{\boldsymbol{s}}} = \bar{\boldsymbol{s}}_e + \bar{\boldsymbol{s}}_{\mathrm{q}}$, and $\bar{\boldsymbol{s}}_{\mathrm{q}} = \bar{H}\boldsymbol{q}$. The new $\tilde{R}'_m = \tilde{S}'_m \tilde{S}'^H_m$ is then used for error localization as detailed in Section III. Then, similar to (19) it can be seen that the new optimal m is $\lceil \frac{d'}{2} \rceil$; i.e., we can improve error localization since there are $\lceil \frac{d'}{2} \rceil - \nu \ge \lceil \frac{d}{2} \rceil - \nu$ error-locator polynomials. Hence, the presence of quantization does not really affect the algorithm.

Remark 1. Similar to the subspace approach [14], the extended subspace approach can be applied to the DCT and DST codes; further, it can be used even for the non-BCH DCT and DST codes [15].[4]

Remark 2. Knowing that $\tilde{R}_m$ can also be used to determine the number of errors ν [21], where the extended error localization is applicable, $\tilde{R}'_m$ can be used for this purpose and it improves the results reasonably.

Remark 3. Once the location of errors are determined, it is rather simple to find their amplitude. Let H_e denote the matrix consisting of the columns of H corresponding to error indices. Then, errors magnitudes $\boldsymbol{E} = [e_{i_1}, \ldots, e_{i_\nu}]^T$ can be computed from $H_e\boldsymbol{E} = \boldsymbol{s}$, using a least squares method, for example. More details are available in [14, Section 5.3], [19], so we will not discuss it more in depth in this paper.

Before moving on to the next section, we look at extended subspace error localization for a special, yet important class of DFT codes in channel coding where $n = 2k$. For such a code, $d = k$ and X_p^d is $+1$ (-1) for errors in the even (odd) positions in the codewords. Then, if all errors are in the even (odd) positions, we can simply replace $\bar{\boldsymbol{s}}$ with $\boldsymbol{s}$ $(-\boldsymbol{s})$. Thus, using (31) and (26) we can form $S'_m(\tilde{S}'_m)$ and the corresponding $R'_m(\tilde{R}'_m)$ for $d' = 2d$. Subsequently, the eigendecomposition of $\tilde{R}'_m$ for $m = \lceil d'/2 \rceil$ increases the number of polynomials in $\mathcal{F}$ and their degree. This can lead to substantial improvement in error localization; for one thing, Fig. 1 in [25] represents the merit of the extended error localization to the existing one in a $(10, 5)$ code. Such a significant improvement in error localization is achieved by using the same d syndrome samples but forming a larger syndrome matrix which allows a larger noise subspace.

V. Generalized Subspace Decoding

This section is primarily focused on the generalization of the subspace-based decoding of BCH-DFT codes. Meanwhile, the proposed algorithm gives rise to a more general encoding for this class of codes. Let $V_m^{[i]}$ be the matrix whose columns are the ith powers of the *error-locator vectors* of order m, i.e.,

$$V_m^{[i]} = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ X_1^i & X_2^i & \cdots & X_\nu^i \\ \vdots & \vdots & \ddots & \vdots \\ X_1^{i(m-1)} & X_2^{i(m-1)} & \cdots & X_\nu^{i(m-1)} \end{bmatrix}. \tag{34}$$

Also, let $\mathcal{P}_n$ be the set of positive integers less than or equal to n that are *relatively prime* to n. The cardinality of $\mathcal{P}_n$, also called the *Euler phi function* $\phi(n)$, is obviously upper bounded by $|\mathcal{P}_n| = \phi(n) \le n - 1$ where the upper bound is achieved when n is a *prime number*. [5]

Proposition 1. *For any $i \in \mathcal{P}_n$ and $m \ge \nu$, the rank of $V_m^{[i]}$ is equal to the number of errors ν.*

Proof: Since $V_m^{[i]}$ has the form of the Vandermonde matrix with elements $X_1^i, X_2^i, \ldots, X_\nu^i$, to prove the claim it suffices to show that the above elements are distinct for any

[4]The idea of syndrome extension has been recently applied for error correction of DCT codes, with a different algorithm in [24].

[5]Interested readers may refer to [26, Section 6.6] for Euler's function and nth roots of unity.

$i \in \mathcal{P}_n$. Suppose for contradiction that $X_p^i = X_{p'}^i$ for some $p \neq p'$, $p, p' \in [1, \ldots, \nu]$, and $i \in \mathcal{P}_n$; thus $e^{\frac{j2\pi i}{n}(i_p - i_{p'})} = 1$ which implies $n = \frac{i}{l}(i_p - i_{p'})$ for some integer l. This means that n and i are not relatively prime, which is contradicting. ∎

Next, for any $i \in \mathcal{P}_n$ and $m > \nu$, the columns of $V_m^{[i]}$ define a ν-dimensional subspace of the m-dimensional vector space which we will refer to as the channel error subspace. We will shortly prove that, for an (n, k) code, this generalizes the notion of [13] by introducing $\phi(n)$ different sets of *spanning basis* for channel error subspace rather than just one. Recall from linear algebra that there are infinitely many different sets of basis which can span the same subspace. Clearly, the channel error subspace of (15) is attained for $i = 1$. Initially, similar to (12), for $\nu + 1 \leq m \leq d - \nu + 1$, we define the ith syndrome matrix by

$$S_m^{[i]} = V_m^{[i]} D V_{d-m+1}^{[i]\,T}. \tag{35}$$

It can be seen that $S_m^{[i]}$, for each i, is composed of d syndrome samples[6] at most; these samples can be identified in the first row and last column. Before proceeding, we should determine the entries of $S_m^{[i]}$. Through simple algebra, one can show that the right-hand side of (35) simplifies to

$$S_m^{[i]} = \begin{bmatrix} s_{\lceil 0 \rceil_n} & s_{\lceil i \rceil_n} & \cdots & s_{\lceil i(d-m) \rceil_n} \\ s_{\lceil i \rceil_n} & s_{\lceil 2i \rceil_n} & \cdots & s_{\lceil i(d-m+1) \rceil_n} \\ \vdots & \vdots & \ddots & \vdots \\ s_{\lceil i(m-1) \rceil_n} & s_{\lceil im \rceil_n} & \cdots & s_{\lceil i(d-1) \rceil_n} \end{bmatrix}, \tag{36}$$

in which $i \in \mathcal{P}_n$ and the subscripts are interpreted modulo n such that $\lceil a \rceil_n \triangleq b + 1$ where $a \equiv b \pmod{n}$ and $0 \leq b < n$. To see the rationale behind the modulo operation recall that $X_p^n = 1$ for any p. Again, it is easy to see that $S_m^{[1]} = S_m$. However, note that for $i > 1$ we will have a different syndrome matrix than (12), where the syndrome samples and their order varies based on i. Note that, the elements of $S_m^{[i]}$, in general, are given by

$$s_\ell = \frac{1}{\sqrt{n}} \sum_{p=1}^{\nu} e_{i_p} X_p^{\alpha - 1 + \ell}, \quad \ell = 1, \ldots, n. \tag{37}$$

Thus, its elements are available, from (8), for $\ell = 1, \ldots, d$, and for $\ell = d + 1, \ldots, n$ they become equal to $\bar{s}_{\ell - d}$, where $\bar{s}_\ell$ is the extended syndrome sample defined in (30). Once the elements of $S_m^{[i]}$ are properly set, we can define the ith covariance matrix

$$R_m^{[i]} = S_m^{[i]} S_m^{[i]\,H}, \tag{38}$$

and eigendecompose it as

$$R_m^{[i]} = [U_{\mathrm{e}}^{[i]}\ U_{\mathrm{q}}^{[i]}] \begin{bmatrix} \Delta_{\mathrm{e}}^{[i]} & \mathbf{0} \\ \mathbf{0} & \Delta_{\mathrm{q}}^{[i]} \end{bmatrix} [U_{\mathrm{e}}^{[i]}\ U_{\mathrm{q}}^{[i]}]^H. \tag{39}$$

[6]With a little abuse of notation, we use the term syndrome samples both for syndrome and extended syndrome samples. Thus, we assume we have s_ℓ in which ℓ can take any values from 1 to n, whereas originally this range was $[1, \ldots, d]$. Such a case is plausible, for instance, in rate-adaptive systems based on DFT codes, as discussed in Section VII.

The matrices $U_{\mathrm{e}}^{[i]}, U_{\mathrm{q}}^{[i]}, \Delta_{\mathrm{e}}^{[i]}$, and $\Delta_{\mathrm{q}}^{[i]}$, respectively, have the same sizes as $U_{\mathrm{e}}, U_{\mathrm{q}}, \Delta_{\mathrm{e}}$, and Δ_{q} and hold similar properties. Specifically, we have

Proposition 2. *The columns of $U_{\mathrm{e}}^{[i]}$ span the channel error subspace.*

Proof: On the one hand, we observe that the rank of $R_m^{[i]}$ is ν since the rank of $S_m^{[i]}$ is so by construction. Therefore, $\Delta_{\mathrm{q}}^{[i]} = \mathbf{0}$ and $R_m^{[i]}$ can be expressed as $R_m^{[i]} = U_{\mathrm{e}}^{[i]} \Delta_{\mathrm{e}}^{[i]} U_{\mathrm{e}}^{[i]\,H}$. On the other hand, from (38) and (35), it can be seen that $R_m^{[i]} = V_m^{[i]} D V_{d-m+1}^{[i]\,T} V_{d-m+1}^{[i]\,*} D^H V_m^{[i]\,H}$. Hence, $U_{\mathrm{e}}^{[i]} = V_m^{[i]} M^{[i]}$ where $M^{[i]} = D V_{d-m+1}^{[i]\,T} (\Delta_{\mathrm{e}}^{[i]})^{\frac{-1}{2}}$; i.e., the columns of $U_{\mathrm{e}}^{[i]}$ can be expressed as linear combinations of the columns of $V_m^{[i]}$, and vice versa; this completes the proof. ∎

An immediate implication of the above proposition is that the columns in $U_{\mathrm{q}}^{[i]}$ span the noise subspace. More importantly, we have the following theorem.

Theorem 1. *For an (n, k) BCH-DFT code, defined by* (2)*, there exist $\phi(n)$ syndrome matrices for decoding, where $\phi(n)$ is the Euler phi function.*

Proof: On the one hand, from eigendecomposition in (39) we have $U_{\mathrm{e}}^{[i]\,H} U_{\mathrm{q}}^{[i]} = \mathbf{0}$. On the other hand, from Proposition 2 we know that the columns in $U_{\mathrm{e}}^{[i]}$ and $V_m^{[i]}$ span the same subspace. Therefore,

$$V_m^{[i]\,H} U_{\mathrm{q}}^{[i]} = \mathbf{0} \quad \forall\, i \in \mathcal{P}_n. \tag{40}$$

Hence, since i can take $\phi(n)$ different values, one can make $\phi(n)$ different syndrome matrices $S_m^{[i]}$, as defined in (36), and utilize them for error localization and detection. ∎

The *diversity* introduced in Theorem 1 brings in two main novelties which are discussed in the remainder of this section.

A. Improved Decoding

Since (40) holds for any $i \in \mathcal{P}_n$, for a complex variable x we define $\boldsymbol{v}^i = [1, x^i, x^{2i}, \ldots, x^{i(m-1)}]^T$ and form

$$F^{[i]}(x) \triangleq \sum_{j=1}^{m-\nu} \boldsymbol{v}^{i\,H} \boldsymbol{u}_{q,j}^{[i]} = \sum_{j=1}^{m-\nu} \sum_{k=0}^{m-1} f_{jk}^{[i]} x^{ki}. \tag{41}$$

For each i, the function $F^{[i]}(x)$ can be considered as sum of $m - \nu$ polynomials $\{f_j^{[i]}\}_{j=1}^{m-\nu}$ of order $m-1$; each polynomial corresponds to one column of $U_{\mathrm{q}}^{[i]}$. Let $\mathcal{F}^{[i]}$ denote this set of polynomials. In view of (40), $F^{[i]}(x) = 0$ for $X_1, \ldots, X_\nu$, and these are the only common roots of $\{f_j^{[i]}\}$ over the nth roots of unity; thus, the error locations can be determined by finding the zeros of $F^{[i]}(x)$ over the set of nth roots of unity. That is, each $F^{[i]}(x)$ individually can be employed for error localization. Therefore, we can equivalently define

$$\Gamma(x) \triangleq \sum_{i \in \mathcal{P}_n} F^{[i]}(x) = \sum_{i \in \mathcal{P}_n} \sum_{j=1}^{m-\nu} \sum_{k=0}^{m-1} f_{jk}^{[i]} x^{ki}, \tag{42}$$

and use it for error localization. That is, the zeros of $\Gamma(x)$ over the set of nth roots of unity give $X_1, \ldots, X_\nu$.

Comparing with (18), it can be seen that (42) combines $\phi(n)$ subspace-based decoding functions; this is referred to as *diversity* in this paper. Hence, the decoding polynomial $\Gamma(x)$

introduces both *diversity* and *degrees of freedom*[7] in comparison with the error-locator polynomial $\Lambda(x)$, whereas $F(x)$ provides degrees of freedom only. Obviously, $1 \leq \phi(n) \leq n-1$ and diversity gain $\phi(n)$ is maximized when n is a *prime number*. Remind that the diversity is achieved at the expense of an increase in the number of transmitted syndromes, for a given code, which implies a lower transmission rate.

It should be emphasized that when there is no quantization error, utilizing (42) presents no gain over (18), for the same reason that there is no difference in using (18) over (10). In other words, the *coding-theoretic*, *subspace*, and *generalized subspace* approaches all have the same performance, and result in the exact location of errors, as long as the number of channel errors are within the capacity of the code. Nevertheless, when quantization error comes into play, the generalized subspace approach outperforms the subspace approach and the subspace approach does better than the coding-theoretic one.

Now let us analyze the effect of quantization error. The distorted syndrome matrix can be represented as $\tilde{S}_m^{[i]} = S_m^{[i]} + Q_m^{[i]}$. Then, the eigendecomposition of the corresponding covariance matrix $\tilde{R}^{[i]} = \tilde{S}_m^{[i]} \tilde{S}_m^{[i]\,H}$ results in

$$\tilde{R}_m^{[i]} = [\tilde{U}_{\mathrm{e}}^{[i]}\ \tilde{U}_{\mathrm{q}}^{[i]}] \begin{bmatrix} \tilde{\Delta}_e^{[i]} & \mathbf{0} \\ \mathbf{0} & \tilde{\Delta}_{\mathrm{q}}^{[i]} \end{bmatrix} [\tilde{U}_{\mathrm{e}}^{[i]}\ \tilde{U}_{\mathrm{q}}^{[i]}]^H. \tag{43}$$

Let $\tilde{U}_{\mathrm{e}}^{[i]} = U_{\mathrm{e}}^{[i]} + U_{\mathrm{qe}}^{[i]}$ and $\tilde{U}_{\mathrm{q}}^{[i]} = U_{\mathrm{q}}^{[i]} + U_{\mathrm{qq}}^{[i]}$. Following the same line of arguments as for (40) we obtain

$$V_m^{[i]\,H} \tilde{U}_{\mathrm{q}}^{[i]} = V_m^{[i]\,H} (U_{\mathrm{q}}^{[i]} + U_{\mathrm{qq}}^{[i]}) = V_m^{[i]\,H} U_{\mathrm{qq}}^{[i]} \triangleq \Delta_{\tilde{F}}^{[i]}, \tag{44}$$

and similar to (42) we get

$$\tilde{\Gamma}(x) = \sum_{i \in \mathcal{P}_n} \tilde{F}^{[i]}(x) = \sum_{i \in \mathcal{P}_n} \sum_{j=1}^{m-\nu} \sum_{k=0}^{m-1} \tilde{f}_{jk}^{[i]} x^{ki}. \tag{45}$$

Our goal is to reduce the effect of $\Delta_{\tilde{F}}^{[i]}$ and this is done through adding polynomials for different i since the entries of

$$\Delta_{\tilde{\Gamma}} = \frac{1}{\phi(n)} \sum_{i \in \mathcal{P}_n} \Delta_{\tilde{F}}^{[i]} \tag{46}$$

diminish as the cardinality of $\mathcal{P}_n$ increases. Intuitively, this is because the perturbation caused by the quantization error is reduced by averaging.

B. Generalized Encoding

Another characteristic of the generalized syndrome matrix is that, except for $i = 1$, the syndrome samples used to build (36) are not successive samples. This suggests that $n-k$ zeros in the frequency domain, padded by Σ in (2) to construct a BCH code, are not constrained to be consecutive. In others words, we can have BCH-DFT codes with non-consecutive zeros in the frequency domain. Equivalently, the rows of H are not to be the consecutive powers of the first n powers of ω. This is because for $\xi = \omega^i$, where i is relatively prime to n, $\{\omega^0, \omega^1, \omega^2, \ldots, \omega^{n-1}\}$ and $\{\xi^0, \xi^1, \xi^2, \ldots, \xi^{n-1}\}$ both are the set of roots of unity. This was originally observed by Marshall [1], however by (36) we present the decoding algorithm as well. Hence, unlike BCH codes which are constructed by selecting a sequence of $n-k$ cyclically adjacent frequencies as the parity frequencies, as a corollary of Theorem 1, we have

Corollary 1. *An (n,k) BCH-DFT code can be constructed by selecting any $n-k$ frequencies, spaced by i, $i < n$, as the parity frequencies, as long as $\gcd(n,i) = 1$. To decode such a code, one can use $S_m^{[i]}$ in* (36)*.*

The fact that the parity frequencies (equivalently, the zeros of Σ) are not required to be cyclically adjacent provides substantial flexibility in constructing real/complex DFT codes. The position of these zeros are determined by the indices of syndrome samples used to build up $S_m^{[i]}$. Specifically, for an (n,k) BCH-DFT code, $\Sigma^{[i]}$ has k nonzero elements given by $\Sigma_{\ell_1,\ell_2}^{[i]} = 1$, $\ell_1 = \lceil i(n-\ell_2) + \alpha \rceil_n$ and $\ell_2 = 1, \ldots, k$. The parity-check matrix $H^{[i]}$ is comprised of the columns of the IDFT matrix W_n^H corresponding to these d zeros.

VI. Generalized-Extended Subspace Error Localization

To maximize the diversity gain, we can combine the extended algorithm of Section IV with the generalized algorithm of Section V. Suppose the total number of syndrome samples at the decoder is $\bar{d} \in [d, n]$. Then, for each d', $d' = d, \ldots, \bar{d}$, $i \in \mathcal{P}_n$, and $\nu + 1 \leq m \leq d' - \nu + 1$ we define $S_m^{'[i]}$ similar to $S_m^{[i]}$ in (36) as

$$S_m^{'[i]} = \begin{bmatrix} s_{\lceil 0 \rceil_n} & s_{\lceil i \rceil_n} & \cdots & s_{\lceil i(d'-m) \rceil_n} \\ s_{\lceil i \rceil_n} & s_{\lceil 2i \rceil_n} & \cdots & s_{\lceil i(d'-m+1) \rceil_n} \\ \vdots & \vdots & \ddots & \vdots \\ s_{\lceil i(m-1) \rceil_n} & s_{\lceil im \rceil_n} & \cdots & s_{\lceil i(d'-1) \rceil_n} \end{bmatrix}, \tag{47}$$

with the optimal value of $m = \lceil d'/2 \rceil$. Next, we can compute

$$R_m^{'[i]} = S_m^{'[i]} S_m^{'[i]\,H}, \tag{48}$$

and eigendecompose it as

$$R_m^{'[i]} = [U_{\mathrm{e}}^{'[i]}\ U_{\mathrm{q}}^{'[i]}] \begin{bmatrix} \Delta_{\mathrm{e}}^{'[i]} & \mathbf{0} \\ \mathbf{0} & \Delta_{\mathrm{q}}^{'[i]} \end{bmatrix} [U_{\mathrm{e}}^{'[i]}\ U_{\mathrm{q}}^{'[i]}]^H. \tag{49}$$

Again $U_{\mathrm{e}}^{'[i]}$ and $\Delta_{\mathrm{e}}^{'[i]}$ have the same sizes as U_{e} and Δ_{e}, and like Proposition 2 we have

Proposition 3. *The columns of $U_{\mathrm{e}}^{'[i]}$ span the channel error subspace.*

Proof: The proof is very similar to that of Proposition 2, so we omit it for brevity. Just note that similar to (35), $S_m^{'[i]} = V_m^{[i]} D V_{d'-m+1}^{[i]\,T}$ and thus the rank of $S_m^{'[i]}$ and $R_m^{'[i]}$ is ν. ■

Note that for $i = 1$ we can always form $S_m^{'[i]}$ whereas for other $i > 1$ we may not be able to form $S_m^{'[i]}$, depending on the availability of the corresponding syndrome samples. One can see that we need $s_{\lceil i(d'-j) \rceil_n}$ for any $j = 1, \ldots, d'$ in order to form $S_m^{'[i]}$. Then, since we have assumed that there are $\bar{d}$ syndrome samples at the decoder, we can form $S_m^{'[i]}$ if and only if $\lceil i(d'-j) \rceil_n \leq \bar{d}$ for any $j = 1, \ldots, d'$. Let $\mathcal{I}_m$ be the set of those $i \in \mathcal{P}_n$ for which all elements of $S_m^{'[i]}$ are

[7] *Diversity* and *degrees of freedom* are well-established terms in wireless communication [27]; we use them in this context for the similarity of concepts.

available, for a given $\bar{d}$. It can be seen that $1 \leq |\mathcal{I}_m| \leq \phi(n)$, where the upper bound is attainable if and only if $\bar{d} = n$. Note that $\mathcal{I}_m$ is non-empty as it always contains $i = 1$. Since for each $i \in \mathcal{I}_m$ and each $d' = d, \ldots, \bar{d}$ we have one $S_m^{'[i]}$, we can form the corresponding $F^{'[i]}(x)$, define

$$\psi(x) \triangleq \sum_{m=\lceil \frac{\bar{d}}{2} \rceil}^{\lceil \frac{\bar{d}}{2} \rceil} \sum_{i \in \mathcal{I}_m} F^{'[i]}(x) = \sum_{m=\lceil \frac{\bar{d}}{2} \rceil}^{\lceil \frac{\bar{d}}{2} \rceil} \sum_{i \in \mathcal{I}_m} \sum_{j=1}^{m-\nu} \sum_{k=0}^{m-1} f_{jkm}^{'[i]} x^{ki}, \tag{50}$$

and use it to find the location of errors. For each d' we use the corresponding optimal $m = \lceil d'/2 \rceil$. Then, comparing (50) with (18), we define the total decoding diversity as

$$\mathcal{D}(d, \bar{d}) = \sum_{m=\lceil d/2 \rceil}^{\lceil \bar{d}/2 \rceil} |\mathcal{I}_m|, \tag{51}$$

considering the two extreme cases, i.e., $\bar{d} = d$ and $\bar{d} = n$, it can be checked that, $1 \leq \mathcal{D}(d, \bar{d}) \leq (\lceil \frac{n}{2} \rceil - \lceil \frac{n-k}{2} \rceil + 1)\phi(n)$. The upper bound can be achieved only if n syndrome samples are available at the decoder. In such an extreme case, $\mathcal{I}_m = \mathcal{P}_n$ thus $|\mathcal{I}_m| = \phi(n)$ and Theorem 1 can be generalized as

Theorem 2. *In addition to $\phi(n)$ syndrome matrices, one can define $(\lceil \frac{n}{2} \rceil - \lceil \frac{n-k}{2} \rceil)\phi(n)$ extended syndrome matrices of different sizes for decoding an (n, k) BCH-DFT code.*

Finally, it is worth mentioning that, by combining the extended and generalized approaches, when there are $\bar{d}$ syndrome samples to decode ν errors, the total number of error-locator polynomials is given by

$$\mathcal{G} = \sum_{m=\lceil d/2 \rceil}^{\lceil \bar{d}/2 \rceil} |\mathcal{I}_m|(m - \nu). \tag{52}$$

What we discussed in this section was for unquantized DFT codes. For quantized codes we will have $\tilde{S}_m^{'[i]}, \tilde{R}_m^{'[i]}, \tilde{F}^{'[i]}(x)$ and $\tilde{\psi}(x)$. For such codes, by adding many ($\mathcal{G}$) polynomials each of which may have different roots due to quantization error, $\tilde{\psi}(x)$ can result in more accurate roots and improve error localization.[8] Then $\mathcal{G}$ can be an indicator of the generalized-extended subspace-based error localization gain with respect to the coding-theoretic approach. In the extreme case where $\bar{d} = n$ and n is a prime number $\mathcal{G}$ is huge.

VII. Application and Examples

Distributed *lossless* compression of two correlated sources can be as efficient as their joint compression [28]. This is also valid for *lossy* source coding with side information at the decoder for jointly Gaussian sources and the mean-squared error (MSE) distortion measure [29]. Typically, DSC is done by quantizing the sources and applying Slepian-Wolf coding in the binary domain. Slepian-Wolf coding can be implemented in the analog domain as well [6] which outperforms its binary counterpart for certain scenarios, e.g., an impulsive correlation model. The proposed DSC schemes based on DFT codes, both parity and syndrome approaches, are also appropriate for low-delay coding as they perform sufficiently well even when short source blocks are encoded [21].

[8]Thus far, it should be clear that any of the error locator polynomials in (10), (18), (42), and (50), give the exact location of errors; i.e., they have the same performance.

A. Rate-Adaptive Distributed Lossy Source Coding

When the statistical dependency between the sources varies or is not known at the encoder, a *rate-adaptive* system with feedback is an appealing solution [30]. Rate-adaptive DSC based on binary codes, e.g., *puncturing* the parity or syndrome bits of turbo and LDPC codes, have been proposed in [30], [31]. In the sequel, we extend DSC based on DFT codes [6] to perform DSC in a rate-adaptive fashion. We consider two continuous-valued correlated sources $\boldsymbol{x}$ and $\boldsymbol{y}$ where x_i and y_i are statistically dependent by $y_i = x_i + e_i$, and e_i is continuous, i.i.d., and independent of x_i. The goal is to compress $\boldsymbol{x}$ given that $\boldsymbol{y}$ is known at the decoder, only.

Rate-adaption using puncturing is not natural for syndrome-based DSC systems [31]. Instead, the encoder can transmit a short syndrome based on an aggressive code and augment it with additional syndrome samples, if decoding fails. This process loops until the decoder gets sufficient samples for successful decoding. This approach is viable only for feedback channels with reasonably short round-trip time [30].

1) Syndrome-based approach: In the syndrome-based DSC based on DFT codes, as one can find in [6, Fig. 2], the encoder transmits $\boldsymbol{s}_x = H\boldsymbol{x}$ to the decoder. At the decoder, we have access to the side information $\boldsymbol{y} = \boldsymbol{x} + \boldsymbol{e}$ and we can compute its syndrome so as to find $\boldsymbol{s}_e = \boldsymbol{s}_y - \boldsymbol{s}_{\boldsymbol{x}}$. Then, based on this syndrome samples, one can form the syndrome matrix and correct the errors, as explained in Section III.

For rate adaptation, if required, the encoder transmits $\bar{\boldsymbol{s}}_x = \bar{H}\boldsymbol{x}$ sample by sample; likewise, the receiver can compute $\bar{\boldsymbol{s}}_y = \bar{H}\boldsymbol{y} = \bar{\boldsymbol{s}}_x + \bar{\boldsymbol{s}}_e$ and evaluate $\bar{\boldsymbol{s}}_e = \bar{\boldsymbol{s}}_y - \bar{\boldsymbol{s}}_x$. After that, we can form the extended or generalized syndrome matrices where $\boldsymbol{s} = \boldsymbol{s}_e$, $\bar{\boldsymbol{s}} = \bar{\boldsymbol{s}}_e$, and

$$s_\ell = \begin{cases} s_\ell, & 1 \leq \ell \leq d, \\ \bar{s}_{\ell-d}, & d < \ell \leq n. \end{cases} \tag{53}$$

and use it for decoding. In short, the rate adaptation algorithm can be summarized as:

- The decoder requests some extra syndrome samples based on the estimated number of errors, for example when $\hat{\nu} > t$ where $\hat{\nu}$ is the estimated number of errors.
- The encoder computes $\bar{\boldsymbol{s}}_x = \bar{H}\boldsymbol{x}$ and transmits it to the decoder sample by sample.
- The decoder computes $\bar{\boldsymbol{s}}_y = \bar{H}\boldsymbol{y}$ and finds $\bar{\boldsymbol{s}}_e = \bar{\boldsymbol{s}}_y - \bar{\boldsymbol{s}}_x$. It then can form R'_m and use the extended subspace decoding algorithms to find the location of errors.

As usual, when quantization is considered this equation needs to be updated as

$$\tilde{s}'_\ell = \begin{cases} \tilde{s}_\ell, & 1 \leq \ell \leq d, \\ \tilde{\bar{s}}_{\ell-d}, & d < \ell \leq n, \end{cases} \tag{54}$$

in which $\tilde{\boldsymbol{s}} = \boldsymbol{s}_e + \boldsymbol{s}_{\mathrm{q}}$, $\tilde{\bar{\boldsymbol{s}}} = \bar{\boldsymbol{s}}_e + \bar{\boldsymbol{s}}_{\mathrm{q}}$, and $\bar{\boldsymbol{s}}_{\mathrm{q}} = \bar{H}\boldsymbol{q}$. The new $\tilde{R}'_m = \tilde{S}'_m \tilde{S}'^H_m$ then is used for error localization as detailed in Sections III–V. Note that the code is incremental, so the encoder does not need to re-encode the sources when more samples are requested. It buffers and transmits the extra syndrome samples to the decoder sample by sample.

2) Parity-based approach: In the parity-based DSC based on DFT codes [6, Fig. 3], for an input sequence $\boldsymbol{x}$, the encoder computes the codeword $\boldsymbol{c} = [\boldsymbol{x} \mid \boldsymbol{p}]^T$ with respect to a systematic DFT code and transmits only $\boldsymbol{p}$, for the sake of compression. At the decoder, we have access to $\boldsymbol{y} = \boldsymbol{x}+\boldsymbol{e}$ (the noisy version of $\boldsymbol{x}$) in addition to $\boldsymbol{p}$; we form $\boldsymbol{z} = [\boldsymbol{y} \mid \boldsymbol{p}]^T$ and compute its syndrome. Since $\boldsymbol{z} = \boldsymbol{c} + \boldsymbol{e}'$ where $\boldsymbol{e}' = [\boldsymbol{e} \mid \boldsymbol{0}]^T$ and $\boldsymbol{s}_c = 0$, we have $\boldsymbol{s}_z = \boldsymbol{s}_{e'}$. Thus, we have the syndrome of error and we can perform decoding.

Although parity- and syndrome-based DSC systems present somewhat different methods for binning, the technique we use for rate-adaptation is the same. For rate adaptation in a parity-based system, once requested by decoder, the encoder computes and transmits $\bar{\boldsymbol{s}}_c = \bar{H}\boldsymbol{c}$; the decoder also computes $\bar{\boldsymbol{s}}_z = \bar{H}\boldsymbol{z} = \bar{\boldsymbol{s}}_c + \bar{\boldsymbol{s}}_{e'}$ and evaluate $\bar{\boldsymbol{s}}_{e'} = \bar{\boldsymbol{s}}_z - \bar{\boldsymbol{s}}_c$. The remainder of the algorithm is similar to the syndrome-based rate-adaptive DSC. Observe that even if the parity samples are not error-free, the above algorithm works [21], which gives rate-adaptive distributed joint source-channel coding.

B. Rate-Adaptive Channel Coding

An approach similar to the one used for parity-based rate-adaptive DSC can be used to make rate-adaptive DFT channel codes. There are however a few differences: First, there is no need to use a systematic code for encoding, any DFT code generated by (2) or, in general, as stated in Section V-B can be used for encoding. Second, since the whole codeword is transmitted over a noisy channel, parity samples are no longer error-free. On the other hand, similar to DSC the extended syndrome samples $\bar{\boldsymbol{s}}_c$ are to be sent over a noiseless channel, if an improvement is expected by virtue of the extended and/or generalized subspace decodings.

C. Examples

In order to fully utilize the extended and generalized subspace decoding algorithms, in this subsection, we assume that the decoder knows n syndrome samples of error, for every codeword of an (n, k) code. If there are fewer samples, we cannot build up $S_m^{[i]}$ and $S'^{[i]}_m$ for some i because we may not have all samples corresponding to those matrices. It is worth noting that, once the syndrome samples of error are known, there is now difference in decoding algorithm used for DSC (syndrome- or parity-based) and channel coding problems.

1) Example 1: Consider the $(10, 5)$ code, for which $\mathcal{P}_n = \{1, 3, 7, 9\}$. From (36) we have

$$S_3^{[1]} = \begin{bmatrix} s_1 & s_2 & s_3 \\ s_2 & s_3 & s_4 \\ s_3 & s_4 & s_5 \end{bmatrix}, S_3^{[3]} = \begin{bmatrix} s_1 & s_4 & s_7 \\ s_4 & s_7 & s_{10} \\ s_7 & s_{10} & s_3 \end{bmatrix},$$
$$S_3^{[7]} = \begin{bmatrix} s_1 & s_8 & s_5 \\ s_8 & s_5 & s_2 \\ s_5 & s_2 & s_9 \end{bmatrix}, S_3^{[9]} = \begin{bmatrix} s_1 & s_{10} & s_9 \\ s_{10} & s_9 & s_8 \\ s_9 & s_8 & s_7 \end{bmatrix}.$$

It is seen that different matrices have several samples in common but they differ in some others. The latter implies an increase in the code rate if we wish to exploit more than one matrix for decoding at the same time.

Interestingly, the same syndrome samples are used to form $S_m^{[i]}$ for $i \in \{2, 4, 6, 8\}$. The only difference is that the position of those samples differ for each i, as is seen below

$$S_3^{[2]} = \begin{bmatrix} s_1 & s_3 & s_5 \\ s_3 & s_5 & s_7 \\ s_5 & s_7 & s_9 \end{bmatrix}, S_3^{[4]} = \begin{bmatrix} s_1 & s_5 & s_9 \\ s_5 & s_9 & s_3 \\ s_9 & s_3 & s_7 \end{bmatrix},$$
$$S_3^{[6]} = \begin{bmatrix} s_1 & s_7 & s_3 \\ s_7 & s_3 & s_9 \\ s_3 & s_9 & s_5 \end{bmatrix}, S_3^{[8]} = \begin{bmatrix} s_1 & s_9 & s_7 \\ s_9 & s_7 & s_5 \\ s_7 & s_5 & s_3 \end{bmatrix}.$$

This means that without any increase in code rate, i.e., merely by changing the arrangement of the syndrome samples and summing up the resulting polynomials (averaging the results), the accuracy of the decoding can be improved. Nevertheless, it should be noted that for $i \in \{2, 4, 6, 8\}$ Proposition 1 does not hold because $X_1^i, \ldots, X_\nu^i$ are not distinct anymore. In fact, for such an i, by using $S_3^{[i]}$ the algorithm cannot differentiate the errors in locations i_p and $i_{p'}$ when $i_p \equiv i_{p'} \pmod 5$, because $X_p^i = X_{p'}^i$. That is, the above matrices can be used only if we have some specific side information about errors. For instance, when we know that the first (last) half of the samples in each codeword are error-free. Finally, let us examine

$$S_3^{[5]} = \begin{bmatrix} s_1 & s_6 & s_1 \\ s_6 & s_1 & s_6 \\ s_1 & s_6 & s_1 \end{bmatrix},$$

which is built upon only two syndrome samples. Clearly, the last row and columns can be removed as they are redundant. With $S_3^{[5]}$ one can only tell if the errors are in odd or even locations, since $i_p \equiv i_{p'} \pmod 2 \Rightarrow X_p^5 = X_{p'}^5$.

Remark 4. The last two cases were not included in our generic algorithm in Section V since they cannot be used to determine the location of errors uniquely. However, we can use them to remove the ambiguity partly, as explained above.

2) Example 2: Here, using the $(11, 3)$ code, we explain how we can make use of extending and generalizing the syndrome matrices at the same time. Since $n = 11$ is a prime number, $\mathcal{P}_n = \{1, \ldots, 10\}$ and we can have 10 syndrome matrices for each $d' \in [8, \ldots, 11]$. For the extreme case of $d' = 11$, these matrices share the same elements with different arrangements. For instance,

$$S_6'^{[2]} = \begin{bmatrix} s_1 & s_3 & s_5 & s_7 & s_9 & s_{11} \\ s_3 & s_5 & s_7 & s_9 & s_{11} & s_2 \\ s_5 & s_7 & s_9 & s_{11} & s_2 & s_4 \\ s_7 & s_9 & s_{11} & s_2 & s_4 & s_6 \\ s_9 & s_{11} & s_2 & s_4 & s_6 & s_8 \\ s_{11} & s_2 & s_4 & s_6 & s_8 & s_{10} \end{bmatrix},$$

and

$$S_6'^{[9]} = \begin{bmatrix} s_1 & s_{10} & s_8 & s_6 & s_4 & s_2 \\ s_{10} & s_8 & s_6 & s_4 & s_2 & s_{11} \\ s_8 & s_6 & s_4 & s_2 & s_{11} & s_9 \\ s_6 & s_4 & s_2 & s_{11} & s_9 & s_7 \\ s_4 & s_2 & s_{11} & s_9 & s_7 & s_5 \\ s_2 & s_{11} & s_9 & s_7 & s_5 & s_3 \end{bmatrix}.$$

As we proved in Section VI, both of these matrices, and any of those 10 matrices on the whole, result in the exact location

of errors based on (50). In practice, due to quantization, we have $\tilde{S}_m^{'[i]} = S_m^{'[i]} + Q_m^{'[i]}$. Hence,

$$\begin{aligned}\tilde{R}_m^{'[i]} &= \tilde{S}_m^{'[i]} \tilde{S}_m^{'[i]H} \\ &= R_m^{'[i]} + \underbrace{S_m^{'[i]} Q_m^{'[i]H} + Q_m^{'[i]} S_m^{'[i]H} + Q_m^{'[i]} Q_m^{'[i]H}}_{T_m^{'[i]}(Q_m^{'[i]})} \end{aligned} \quad (55)$$

From Proposition (3) we know that $R_m^{'[i]}$, for any $i = 1, \ldots, 10$, results in the exact location of errors. However, for quantized DFT codes, $\tilde{R}_m^{'[i]}$ is used for decoding and it yields slightly different results for each i, due to different combinations of the quantization error at the decoding process which is represented by $T_m^{'[i]}$. This brings in some sort of decoding *diversity*, and we exploit this diversity to improve the decoding accuracy, as explained in Sections VI and V.

VIII. Simulation Results

To evaluate the performance of the proposed algorithms we perform simulations using a Gauss-Markov source X, with a mean zero, a variance one, and a correlation coefficient $\rho = 0.9$. The Gauss-Markov process $\{X_i\}$ is generated based on the following recursion

$$X_i = \sqrt{1-\rho^2} Z_i + \rho X_{i-1}, \quad (56)$$

in which $\{Z_i\}$ is a zero-mean i.i.d. Gaussian process with variance 1, and $0 \leq \rho < 1$ is the correlation coefficient [32]. Given an (n, k) code, the encoder divides X into blocks $\boldsymbol{x}$ of length n and generates the syndrome $\boldsymbol{s}_x$ and extended syndrome $\bar{\boldsymbol{s}}_x$; these are then quantized by a 3-bit uniform quantizer with step size $\Delta = 0.25$. At the decoder, we have the quantized syndromes and the side information $\boldsymbol{y} = \boldsymbol{x} + \boldsymbol{e}$, where $\boldsymbol{e}$ is the error vector. The number of errors in each block is ν, $\nu \leq t$, where t is the error correction capacity of the code. Similar to [13], we assume that the error components are fixed. We plot the relative frequency of correct localization of errors[9] versus channel-error-to-quantization-noise ratio (CEQNR) which is defined as the ratio of channel error power to the quantization noise power, i.e.,

$$\text{CEQNR} \triangleq \frac{\sigma_e^2}{\sigma_{\text{q}}^2}. \quad (57)$$

In Fig. 1, we compare the frequency of correct localization of errors for the subspace and extended subspace approaches given a $(10, 5)$ code for different errors. The gain due to the extended subspace method is remarkable both for one and two errors; it is more significant for two errors. In fact, as discussed in Section III-B, for $\nu = t$ the subspace approach loses its degrees of freedom (DoF) and its performance drops to that of the coding-theoretic approach. Providing some extra DoF, at the expense of a higher code rate, the extended subspace approach considerably improves the error localization. The gain caused by the extended subspace method increases for codes with higher capacity. For instance, simulation results for a $(17, 9)$ DFT code, presented in Fig. 2, show a significant gain in any CEQNR between 10 to 40 dB; this is achieved by sending 5 additional syndrome samples.

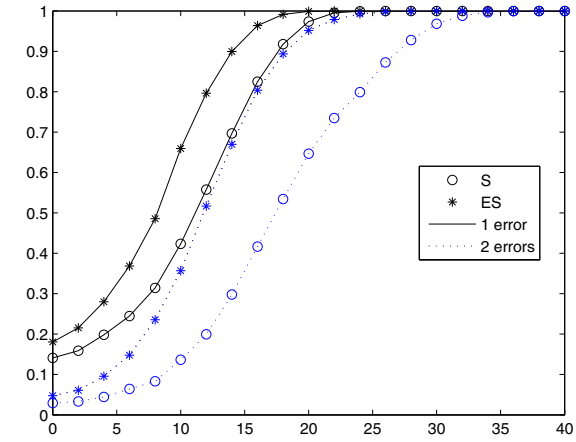

Fig. 1. Probability of error localization in the subspace (S) and extended subspace (ES) approaches at different CEQNRs for a $(10, 5)$ DFT code. The curves for the extended case are based on 2 additional syndrome samples, implying that the code rate is increased from 0.5 to 0.7.

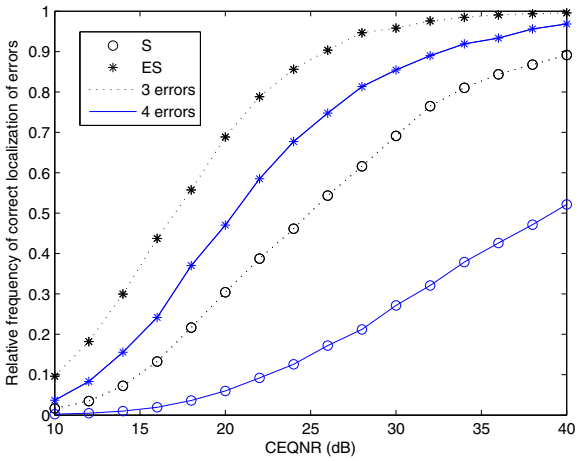

Fig. 2. Probability of error localization in the subspace and extended subspace methods at different CEQNRs for a $(17, 9)$ DFT code. The curves for the extended case are based on 5 additional syndrome samples.

Next, we evaluate the performance of the *generalized subspace* method. To begin with, in Fig. 3 we show the merit of the generalized subspace error localization with respect to the subspace method. An important question is whether the generalized subspace decoding performs better than the extended subspace method. To answer this question, we compare the two algorithms for different codes and various numbers of extended syndrome. In Fig. 4, we compare the frequency of correct localization of errors for the extended and generalized subspace approaches for a $(11, 5)$ code with different number of errors; both methods use 3 syndrome samples more than the subspace method. The same simulation is done for 4 extra

[9]The probability of error localization is defined as relative frequency of correct localization of all ν errors in each block (codeword) to the number of blocks. With this definition, correct error localization will guarantee a decent error correction also. In [13], this parameter is defined as the total number of correctly identified locations to the total number of errors. If such a criterion is used, all curves corresponding to $\nu > 1$ will shift up and we will get better probability of error localization.

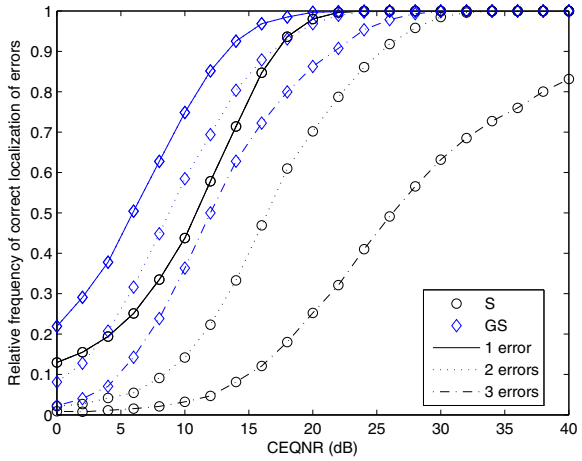

Fig. 3. Probability of correct localization of 1 to 3 errors for a $(11, 5)$ DFT code using the subspace (S) and generalized subspace (GS) methods with 5 additional syndrome samples.

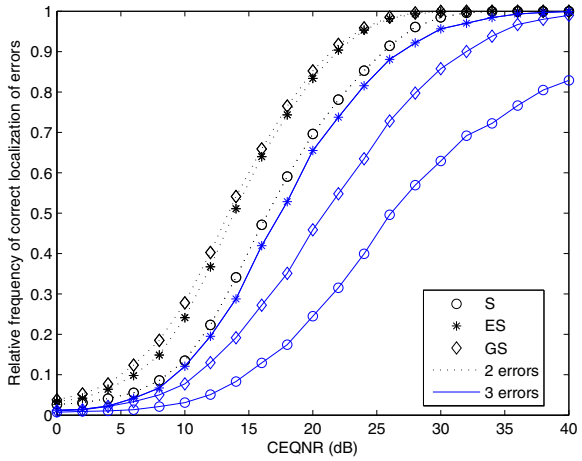

Fig. 4. Probability of correct localization of 2 and 3 errors for a $(11, 5)$ DFT code using the subspace (S), extended subspace (ES), and generalized subspace (GS) methods with 3 additional syndrome samples.

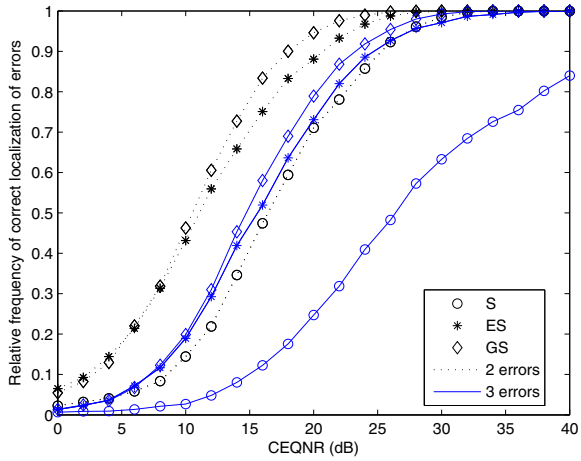

Fig. 5. Probability of correct localization of 2 and 3 errors for a $(11, 5)$ DFT code using the subspace, extended subspace, and generalized subspace methods with 4 additional syndrome samples.

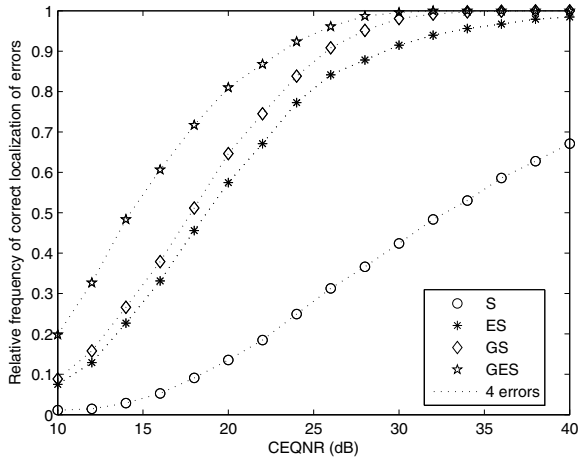

Fig. 6. Probability of correct localization of 4 errors for a $(13, 5)$ DFT code using the subspace, extended subspace, generalized subspace, and generalized-extended subspace (GES) methods with 4 additional syndrome samples.

samples and the results are presented in Fig. 5. It is seen that, for 3 errors, the extended subspace approach produces better results than the generalized subspace method in the first case while the performance gain swaps in the second case. In general, extensive simulations show that the performance of the extended subspace error localization is better than the generalized subspace if just a few extra syndrome samples are available. By increasing the extra samples the gain caused by the generalized subspace method increases sharply such that it can catch up with and even outperform the extended subspace method. In the extreme case, when there are k extra samples, generalized subspace method outperforms its opponent distinctly. The performance of the generalized subspace localization obviously depends on the number of matrices (or equivalently, polynomials); therefore, its performance gain is noticeable when n is prime number.

Finally, let us evaluate the performance of the generalized-extended approach for a couple of codes. We consider a $(13, 5)$ which is capable of correcting up to 4 errors. Since $n = 13$ is a prime number, we expect that the generalized and generalized-extended perform better than the other approaches. This is true as shown in Fig. 6; the gain from generalization is considerably high especially when $\nu \to t$. A similar pattern is seen for other codes, e.g., $(11, 5)$, $(14, 5)$, and $(17, 9)$ codes, to name a few.

In summary, we conclude that no one approach is superior in every situation. The performances of the proposed algorithms vary with $\bar{d}$, ν, t, and CEQNR. However, based on extensive simulation, some of which presented in the paper, the following patterns are observed.

- The generalized and generalized-extended approaches markedly outperform the other ones in different CEQNRs as $\bar{d} \to n$, specially when n is prime (See Fig. 6, for example). This is because $\mathcal{D}$ and thus $\mathcal{G}$ are very large which bring a huge gain.
- When $\bar{d} \to d$, the generalized approach loses its gain as $|\mathcal{I}_m| = 1$. In such cases, the extended subspace is the best. The rationale behind the generalized approach

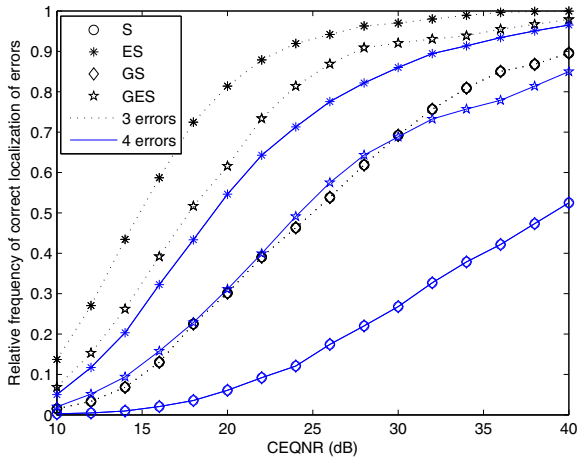

Fig. 7. Probability of correct localization of 3 and 4 errors for a $(17, 9)$ DFT code using different subspace-based error localizations with 4 additional syndrome samples.

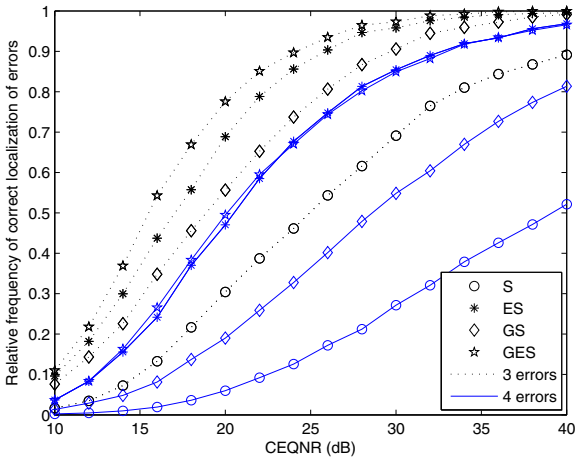

Fig. 8. Probability of correct localization of 3 and 4 errors for a $(17, 9)$ DFT code using different subspace-based error localizations with 5 additional syndrome samples.

is to use multiple syndrome matrices with smaller sizes compared with the extended approaches that use up all syndrome samples in a large syndrome matrix. However, it should be noted that when there are few new syndromes ($\bar{d} \to d$), it is not possible to build up more than one syndrome matrix. In such cases, the generalized approach fails to use extra syndromes while the extended approach uses them up and thus outperforms its opponent. This is examined for a $(17, 9)$ code in Fig. 7 and Fig. 8. As it can be seen, with 4 extra syndrome samples (Fig. 7) the performance of generalized subspace is the same as subspace approach. This is because with that many syndromes we cannot form any generalized syndrome matrix for $i > 1$, therefore there is no gain over the subspace method; in such a case the extended subspace gives the best results, as explained above. But, in Fig. 8, when only one more sample is added the gain from generalization comes in and the generalize-extended method outperforms the extended one.

- The generalized-extended method seems to be the best choice when there are enough extra syndrome samples to build several or more syndrome matrices. This is because it averages as many polynomials as possible to improve the results. As the CEQNR goes up, its dominance is less because the other approaches work well, also.

IX. Conclusions

We have developed three subspace-based algorithms that substantially improve the existing subspace error localization of quantized DFT codes, in favor of extra syndrome samples. The first approach, named extended subspace, simply extends and improves the existing one by increasing the dimension of the quantization noise subspace, or equivalently, the number of polynomials obtainable for error localization. This is followed by another method that generalizes the decoding, and also the encoding, of the DFT codes. We proved that many syndrome matrices, each of which uses d syndromes, can be utilized to decode DFT codes. This diversity is exploited for increasing the decoding accuracy since by averaging the corresponding error locating polynomials the effect of the quantization error diminishes. The third approach, i.e., the generalized-extended subspace decoding, combines the aforementioned algorithms to further increase the decoding gain.

The proposed algorithms are primarily useful for rate-adaptation in a DSC system that uses DFT codes for binning; they can also be used in the channel coding. Rate-adaptation is realized by augmenting the syndrome samples and does not need to re-encode the sources. The extended decoding can be applied to DCT and DST codes, whereas the extension of the generalized subspace decoding to these classes of codes is not straightforward. One possible direction to pursue is to investigate applying this generalized algorithm to DCT and DST codes. Furthermore, constructing any other spanning basis V_m such that the entries of $V_m D V_{d-m+1}^T$ are syndrome samples, with any permutation, is valuable as it can be used, in conjunction with the other spanning matrices developed in this paper, for diminishing the effect of quantization error.

Acknowledgement

The authors would like to thank the anonymous reviewers for their valuable comments and suggestions to improve the presentation of the paper.

References

[1] T. Marshall Jr., "Coding of real-number sequences for error correction: a digital signal processing problem," *IEEE J. Sel. Areas Commun.*, vol. 2, no. 2, pp. 381–392, Mar. 1984.

[2] R. E. Blahut, *Algebraic Codes for Data Transmission*. Cambridge University Press, 2003.

[3] Z. Wang and G. B. Giannakis, "Complex-field coding for OFDM over fading wireless channels," *IEEE Trans. Inf. Theory*, vol. 49, no. 3, pp. 707–720, Mar. 2003.

[4] W. Henkel and F. Hu, "OFDM and analog RS/BCH codes," in *Proc. 2005 International OFDM Workshop.*

[5] A. Gabay, M. Kieffer, and P. Duhamel, "Joint source-channel coding using real BCH codes for robust image transmission," *IEEE Trans. Image Process.*, vol. 16, no. 6, pp. 1568–1583, June 2007.

[6] M. Vaezi and F. Labeau, "Distributed lossy source coding using real-number codes," in *Proc. 2012 IEEE VTC — Fall*, pp. 1–5.

[7] M. F. Duarte and R. G. Baraniuk, "Spectral compressive sensing," *Appl. Comput. Harmon Anal.*, vol. 35, no. 1, pp. 111–129, 2013.

[8] V. K. Goyal, J. Kovačević, and J. A. Kelner, "Quantized frame expansions with erasures," *Appl. Comput. Harmon Anal.*, vol. 10, no. 3, pp. 203–233, 2001.

[9] G. Rath and C. Guillemot, "Frame-theoretic analysis of DFT codes with erasures," *IEEE Trans. Signal Process.*, vol. 52, no. 2, pp. 447–460, Feb. 2004.

[10] B. G. Bodmann and P. K. Singh, "Burst erasures and the mean-square error for cyclic Parseval frames," *IEEE Trans. Inf. Theory*, vol. 57, no. 7, pp. 4622–4635, July 2011.

[11] R. O. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE Trans. Antennas Propag.*, vol. 34, no. 3, pp. 276–280, Mar. 1986.

[12] R. Roy and T. Kailath, "ESPRIT-estimation of signal parameters via rotational invariance techniques," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 37, no. 7, pp. 984–995, 1989.

[13] G. Rath and C. Guillemot, "Subspace algorithms for error localization with quantized DFT codes," *IEEE Trans. Commun.*, vol. 52, no. 12, pp. 2115–2124, Dec. 2004.

[14] ——, "Characterization of a class of error correcting frames for robust signal transmission over wireless communication channels," *EURASIP J. Adv. Signal Process.*, vol. 2005, no. 2, pp. 229–241, 2005.

[15] A. Kumar and A. Makur, "Improved coding-theoretic and subspace-based decoding algorithms for a wider class of DCT and DST codes," *IEEE Trans. Signal Process.*, vol. 58, no. 2, pp. 695–708, Feb. 2010.

[16] S. K. Mitra and Y. Kuo, *Digital Signal Processing: A Computer-Based Approach*. McGraw-Hill, 2006.

[17] R. E. Blahut, *Algebraic Methods for Signal Processing and Communications Coding*. Springer-Verlag, 1992.

[18] M. Vaezi and F. Labeau, "Systematic DFT frames: principle, eigenvalues structure, and applications," *IEEE Trans. Signal Process.*, vol. 61, no. 15, pp. 3774–3885, Aug. 2013.

[19] ——, "Least squares solution for error correction on the real field using quantized DFT codes," in *Proc. 2012 EUSIPCO*, pp. 2561–2565.

[20] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. John Wiley & Sons, 2006.

[21] M. Vaezi and F. Labeau, "Wyner-Ziv coding in the real field based on BCH-DFT codes." Available: http://arxiv.org/abs/1301.0297.

[22] G. A. F. Seber, *A Matrix Handbook for Statisticians*. John Wiley & Sons, 2008.

[23] S. Kay, *Modern Spectral Estimation*. Prentice-Hall, 1988.

[24] G. Redinbo, "Correcting DCT codes with Laurent Euclidean algorithm and syndrome extension," *IEEE Trans. Signal Process.*, vol. 61, no. 9, pp. 2308–2318, May 2013.

[25] M. Vaezi and F. Labeau, "Extended subspace error localization for rate-adaptive distributed source coding," in *Proc. 2013 ISIT*, pp. 2174–2178.

[26] B. L. van der Waerden, E. Artin, and E. Noether, *Algebra, Vol. 1*. Springer, 1966.

[27] D. Tse and P. Viswanath, *Fundamentals of Wireless Communication*. Cambridge University Press, 2005.

[28] D. Slepian and J. K. Wolf, "Noiseless coding of correlated information sources," *IEEE Trans. Inf. Theory*, vol. IT-19, no. 4, pp. 471–480, July 1973.

[29] A. D. Wyner and J. Ziv, "The rate-distortion function for source coding with side information at the decoder," *IEEE Trans. Inf. Theory*, vol. 22, no. 1, pp. 1–10, Jan. 1976.

[30] D. Varodayan, A. Aaron, and B. Girod, "Rate-adaptive codes for distributed source coding," *Signal Process.*, vol. 86, no. 11, pp. 3123–3130, Nov. 2006.

[31] V. Toto-Zarasoa, A. Roumy, and C. Guillemot, "Rate-adaptive codes for the entire Slepian-Wolf region and arbitrarily correlated sources," in *Proc. 2008 ICASSP*, pp. 2965–2968.

[32] T. Wiegand and H. Schwarz, *Source Coding: Part I of Fundamentals of Source and Video Coding*. Now Publishers, 2010.

Mojtaba Vaezi (S'09) is a Ph.D. student in the Electrical and Computer Engineering department at McGill University, Montreal, Canada. He received his B.Sc. and M.Sc. degrees both in Electrical Engineering from Amirkabir University of Technology (Tehran Polytechnic), Tehran, Iran. Mojtaba has served as the president of McGill IEEE Student Branch during 2012-2013.

Before joining McGill in 2009, Mojtaba was the Head of Mobile Radio Network Design and Optimization Department at Ericsson Iran. His research interests are in network information theory and signal processing and includes source and channel coding. Mojtaba is a recipient of a number of academic and leadership awards, including the McGill Engineering Doctoral Award (MEDA) for 2009-2012, and IEEE Larry K. Wilson Regional Student Activities Award in 2013.

Fabrice Labeau is an associate professor in the Electrical and Computer Engineering department at McGill University, Montreal, Quebec, Canada, where he holds the NSERC/Hydro-Qubec Industrial Research Chair in Interactive Information Infrastructure for the Power Grid. He received the Electrical Engineer degree in 1995 from Université catholique de Louvain, Belgium, and the Diplôme d'études spécialisées en Sciences Appliquées, orientation Télécommunications also from UCL in 1996. From 1996 to 2000, he was with the Communications and Remote Sensing Laboratory of UCL. From January to March 1999, he was a visiting scientist to the Signal and Image Department (TSI) of ENST Paris. He received a Ph.D. degree in September 2000 from UCL. His research interests include signal processing applications to e-health and energy management, multirate processing, joint source channel coding, data compression and error-control coding.

He was part of the organizing committee of ICASSP 2004 in Montreal and is/was Technical Program Committee co-chair for the IEEE Vehicular Technology Conference in the Fall of 2006 and 2012, and the IEEE International Conference on Image Processing 2015. He is a Senior Member of IEEE. He currently is the Executive Vice-President of the IEEE Vehicular Technology Society. He has held several administrative and management positions at McGill University, including Associate Department Chair, Associate Dean and Interim Chair. He currently serves as Chair of the Montreal IEEE Signal Processing Society chapter.