Distributed Source-Channel Coding Based on Real-Field BCH Codes

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

***Abstract*—We use** *real-number codes* **to compress statistically dependent sources and establish a new framework for** *distributed lossy source coding* **in which we compress sources before, rather than after, quantization. This change in the order of** *binning* **and** *quantization* **blocks makes it possible to model the correlation between continuous-valued sources more realistically and compensate for the quantization error partially. We then focus on the asymmetric case, i.e., lossy source coding with side information at the decoder. The encoding and decoding procedures are described in detail for a class of real-number codes called discrete Fourier transform (DFT) codes, both for the** *syndrome-* **and** *parity-based* **approaches. We leverage subspace-based decoding to improve the decoding and by extending it we are able to perform distributed source coding in a** *rate-adaptive* **fashion to further improve the decoding performance when the statistical dependency between sources is unknown. We also extend the parity-based approach to the case where the transmission channel is noisy and thus we perform** *distributed joint source-channel coding* **in this context. The proposed system is well suited for low-delay communications, as the mean-squared reconstruction error (MSE) is shown to be reasonably low for very short block length.**

***Index Terms*—Distributed source coding, joint source-channel coding, real-number codes, BCH-DFT codes, syndrome, parity.**

I. Introduction

DISTRIBUTED source coding (DSC) studies compression of statistically dependent sources which do not communicate with each other [1]. As a special case of *lossy* DSC, the *Wyner-Ziv coding* problem [2] considers lossy data compression with side information at the decoder. The current approach to the DSC of continuous-valued sources is to first convert them to discrete-valued sources and then apply *lossless* (Slepian-Wolf) coder [3]–[5]. Similarly, a practical Wyner-Ziv encoder consists of a quantizer and Slepian-Wolf encoder. There are, hence, *quantization* and *binning* losses for the source coder. Despite this, *rate-distortion theory* promises that block codes of sufficiently large length are asymptotically optimal, and they can be seen as vector quantizers followed by fixed-length coders [6]. Therefore, practical *Slepian-Wolf* coders have been realized using different binary channel codes, e.g., low-density parity-check (LDPC) and turbo codes [7]–[9]. These codes, however, are out of the question if low delay is imposed on the system as they may introduce excessive delay when the desired probability of error is very low. The other extreme case, i.e., zero delay source-channel coding, can be achieved through the use of analog mapping [10]–[13]. These schemes have lower complexity but they do not benefit from the advantages of digital communications as they use *analog* communications; they are also far from the theoretical limits.

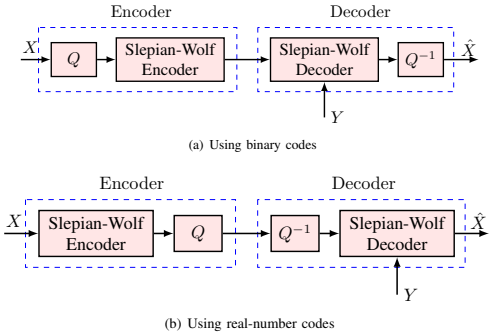

Fig. 1. The Wyner-Ziv coding based on binary and real-number codes. Both schemes can be simply extended to distributed source coding.

In this paper, we establish a new framework for distributed lossy source coding over *digital* communication channel.[1] We propose to first compress the continuous-valued sources and then quantize them, as opposed to the conventional approach. The new framework is compared against the existing one in Fig. 1. It introduces the use of *real-number codes* (see, e.g., [14]–[18]) to represent correlated sources with fewer samples in the real field.

To achieve compression, we send *syndrome* or *parity* samples of the input sequence using a real-number channel code, similar to what is done to compress a binary sequence of data using binary channel codes. Then, we quantize the syndrome or parity samples and transmit them. There are still binning and quantization losses; however, since coding is performed before quantization, error correction is performed in the real field and quantization error can be corrected if two sources are the same over a block of code. A second and more significant advantage of this approach is the fact that the correlation channel model can be more realistic as it captures the dependency between the continuous-valued sources rather than quantized

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 IEEE VTC-Fall, Québec city, Canada, September 2012 [41], and IEEE DSP/SPE Workshop in Napa, California, August 2013 [37].

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 XXXXXXX

[1]The proposed framework can be exploited in analog communication systems simply by removing the quantization block.

sources. In the conventional approach, it is implicitly assumed that quantization of correlated signals results in correlated sequences in the binary domain; this may not necessarily be precise due to the nonlinearity of the quantization operation. To avoid any loss due to the inaccuracy of correlation model, we exploit the correlation between the continuous-valued sources before quantization. The new approach is also capable of alleviating the quantization error. This is possible because coding precedes quantization. Specifically, we use the *Bose-Chaudhuri-Hocquenghem* (BCH) DFT codes [14], [19]–[23] for compression, and owing to them the loss due to quantization can be decreased by a factor of $\frac{k}{n}$ for an (n, k) code [23]–[25]. Additionally, reconstruction loss becomes zero if the two sources are the same over one short codeword of a DFT code. This is achieved in view of modeling the correlation between the two sources in the continuous domain. Moreover, the new framework is suitable for low-delay communications since, by using short DFT codes, a reasonably low reconstruction error is attainable.

As another contribution of this paper, we use a single DFT code both to compress and protect sources against channel variations. This extends the Wyner-Ziv coding to the case where errors can occur during transmission and proposes joint source-channel coding (JSCC) with side information at the decoder, within the realm of real-number codes. This scheme maps short source blocks into channel blocks, and thus it is well suited to low-delay coding.

Finally, by using the extended subspace decoding [26], we propose rate-adaptive DSC both for the syndrome and parity approaches. To do so, the encoder transmits a short syndrome, based on an (n, k) DFT code, and augments it with additional samples if the decoder request. There is no need to re-encode the sources once new samples are requested; the algorithm requires a feedback channel though. Rate-adaptation improves the source rate-distortion especially for short block lengths or when the degree of statistical dependence varies.

Without further estimation block after Slepian-Wolf coding, the MSE performance of DSC and distributed JSCC (DJSCC) systems based on binary codes is limited to the quantization error level whereas the proposed DSC and DJSCC schemes break through this limit. However, if the final MSE is based on a joint estimation error between the quantized source and the decoder side information, it can be smaller than the quantization error in the former cases. Thus, the quantization error is not be a fair benchmark in those cases. To evaluate the effectiveness of the new framework, the MSE is also compared against asymptotic information-theoretic bounds available in the literature.

The paper is organized as follows. In Section II, we motivate the new framework for lossy DSC. In Section III, we study the encoding and decoding of BCH-DFT codes, and we adapt the subspace error localization to the Slepian-Wolf coder. Then, in Section IV, we present the encoder and decoder for the Wyner-Ziv coding based on the BCH-DFT codes, both for the syndrome and parity approaches. The system is extended to the noisy channel case and rate-adaptive coding in Section V and Section VI, respectively. Section VII presents the simulation results and Section VIII provides our concluding remarks.

For notation, we use upper-case and boldface lower-case letters for vectors, boldface upper-case letters for matrices, $(.)^T$ for transpose, $(.)^H$ for conjugate transpose, $(.)^*$ for conjugate, and $\mathrm{tr}(.)$ for the trace. The dimensions of matrices are indicated by one and two subscripts, when required.

II. Real-Number Codes for DSC

Similar to error correction in finite fields, the basic idea of error correcting codes in the *real field* is to insert *redundancy* to a message vector to convert it to a longer vector, called the codeword. However, the insertion of redundancy is done in the real field, i.e., before quantization and entropy coding [14], [15]. One main advantage of *soft redundancy* (real field codes) over *hard redundancy* (binary field codes) is that by using soft redundancy one can go beyond the quantization error level and thus reconstruct continuous-valued signals more accurately.

We introduce the use of real-number codes in lossy compression of correlated signals. As one thing, the proposed system for the Wyner-Ziv coding is depicted in Fig. 1. Although it consists of the same blocks as the existing Wyner-Ziv coding scheme [3], [4], the order of these blocks is changed. Hence, binning is performed before quantization and can benefit from soft redundancy; we use DFT codes for this purpose.

A. Motivations

The change in the order of binning and quantization blocks and the use of DFT codes for binning are motivated by:

1) Realistic correlation model: In the existing framework for lossy DSC, correlation between two sources is modeled after quantization, either using binary [3]–[5], [9], [27] or non-binary [28], [29] correlation models. Admittedly though, due to the nonlinearity of the quantization operation, correlation model between the quantized signals cannot be as accurate as that of the continuous signals. This motivates us to investigate a method that exploits correlation between continuous-valued sources to perform DSC.

2) Alleviating the quantization error: In lossy data compression with side information at the decoder, soft redundancy, added by DFT codes, can be used to correct both quantization errors and (correlation) channel errors. Thus, the loss due to quantization can be recovered, at least partly if not wholly. In fact, if the two sources are exactly the same over a codeword,[2] the quantization error can be compensated for. That is, perfect reconstruction is attainable over the corresponding samples. The loss due to the quantization error is decreased by a factor of code rate ($\frac{k}{n}$ for an (n, k) code) even if correlation is not perfect, i.e., when errors exist. This is because DFT codes are *tight* frames; hence, they minimize the MSE [23]–[25].

3) Low-delay communications: Limited delay is a key constraint in many modern applications, and low-delay coding has recently drawn a lot of attention. If communication is subject to low-delay constraints, the performance of turbo and LDPC codes may not be satisfactory due to the imposed short code length. Early works on DSC are mainly based on

[2]Note that this can happen in practice especially if the code length is short; for example, it is possible when the readings of two closely-located sensors are the same for a short period of time, i.e., over one block of code.

turbo and LDPC codes. Low-delay systems can be realized by mapping short source blocks into channel blocks, in a linear or non-linear fashion [10], [30], [31]. Whether the low-delay requirement exists or not depends on the specific applications. However, even in the applications in which low-delay transmission is not imperative, it is sometimes useful to consider low-dimensional systems for their lower computational complexity. We use DFT codes with short block length and scalar quantization which is suitable if limited delay is required.

B. Correlation Channel Model

Accurate modeling of the correlation between the sources plays a crucial role in the performance evaluation and efficiency of the DSC systems. Existing works on the DSC model the correlation between the continuous-valued sources after quantization. Some of them, mainly theoretical ones, assume that this statistical dependency can be modeled by a binary symmetric channel (BSC) [3]–[5], [9], [27]. Many practical works, such as [28], [29], however, consider non-binary correlation models. In any of those cases, the correlation is modeled after quantization and, due to the nonlinearity of quantization operation, such a correlation model may not be as accurate as a model in the continuous domain. This issue can be dealt with by exploiting the correlation between the continuous-valued sources before quantization.[3]

The correlation between the analog sources X and Y, in general, can be defined by

$$Y = X + E, \tag{1}$$

where E is a real-valued random variable. This model, in which the correlation noise E is independent from X, will be referred to as the *forward correlation channel*.[4] Particularly, the above model represents some well-known models motivated in video coding and sensor networks. Let

$$E \sim \begin{cases} \mathcal{N}(0,\sigma_0^2) & \text{w.p.} \quad p_0, \\ \mathcal{N}(0,\sigma_1^2) & \text{w.p.} \quad p_1, \\ 0 & \text{w.p.} \quad 1-p_0-p_1, \end{cases} \tag{2}$$

in which $\sigma_1^2 = \sigma_i^2 + \sigma_0^2$, $\sigma_i^2 \gg \sigma_0^2$, and $p_0 + p_1 \leq 1$. This is a mixture of Gaussian *impulses* with power σ_i^2 and a *background noise* with power σ_0^2. Then, for $p_0 = 1$ or $p_1 = 1$ the Gaussian correlation is obtained. Further, for $p_0 + p_1 = 1$ the Gaussian-Bernoulli-Gaussian (GBG) and for $p_0 + p_1 < 1$, $p_0 p_1 = 0$ the Gaussian-Erasure (GE) models are realized. The Gaussian model is broadly used in the sensor networks literature whereas the latter two models are more suitable for video applications.

[3]To be precise, the term "correlation channel" is used to represent a statistical dependency rather than a correlation. With this view, it is natural to consider the statistical dependency of analog sources in the analog domain, e.g., as in (1).

[4]Alternatively, one may use $X = Y + E$ to show the dependency [32]. In such a case, the source sample X can be interpreted as the sum of the side information (Y) and of an innovation component (E). This model which is preferred in the distributed video coding literature will be referred to as the *reverse correlation channel*.

The GBG model can be considered as an extension of the jointly Gaussian model. In spite of the Gaussian model, the rate-distortion region of this model is not known. Therefore, its characterization can only be obtained through *lower and upper bounds*. To find such bounds, which will be later used in this paper, one can make use of the rate-distortion function of related, simpler coding problems. We know that when side information Y is available both at the encoder and decoder, the rate-distortion function for jointly Gaussian sources is given by

$$R_{X|Y}(D) = \begin{cases} \frac{1}{2}\log_2\left(\frac{\sigma_{X|Y}^2}{D}\right) & \text{if } 0 \leq D \leq \sigma_{X|Y}^2 \\ 0 & \text{if} \quad D > \sigma_{X|Y}^2 \end{cases} \tag{3}$$

where $\sigma_{X|Y}^2 = \frac{\sigma_X^2\sigma_E^2}{\sigma_X^2+\sigma_E^2}$ for forward correlation channel and $\sigma_{X|Y}^2 = \sigma_E^2$ for reverse correlation channel. This is known as *conditional* source coding problem, and its rate region is obviously a lower bound for lossy source coding with side information available "only" at the decoder (the Wyner-Ziv problem); i.e., $R_{X|Y}^{\mathrm{WZ}}(D) - R_{X|Y}(D) \geq 0$. Yet, Wyner and Ziv [2] proved the intriguing result that for jointly Gaussian memoryless sources and mean-squared error (MSE) distortion $R_{X|Y}^{\mathrm{WZ}}(D) = R_{X|Y}(D)$. [5]

In the following, $R_{X|Y}(D)$ is used to develop lower and upper bounds for $R_{X|Y}^{\mathrm{GBG}}(D)$. Assuming that the position of impulses are revealed both to the encoder and decoder, the rate-distortion function can be obtained in a time division manner. Thus, for $D \leq \sigma_0^2$ we have

$$R_{X|Y}^{\mathrm{GBG}}(D) \geq \underline{\mathrm{R}}_{X|Y}^{\mathrm{GBG}}(D) = \sum_j p_j R_{X|Y,s_j}(D), \tag{4}$$

where $\sigma_{X|Y,s_j} = \sigma_j$, and $j \in \{0,1\}$.[6] It is also straightforward to see that

$$R_{X|Y}^{\mathrm{GBG}}(D) \leq \bar{R}_{X|Y}^{\mathrm{GBG}}(D) = R_{X|Y,s_1}(D). \tag{5}$$

By limiting the knowledge of the position of impulses to the decoder only, one may come up with tighter bounds. This is an interesting topic per se, but it is out of the scope of this work. Yet, to provide another benchmark level, we will evaluate the performance of the proposed systems when the position of impulses are known at the decoder.

We also study the Gauss-Markov sources as in many applications there is dependency between source samples. The rate distortion function of a Gauss-Markov process with a variance σ^2 and a correlation coefficient $0 \leq \rho < 1$ is given by [34]

$$R(D) = \frac{1}{2}\log_2\left(\frac{(1-\rho^2)\sigma^2}{D}\right) \qquad \text{if} \quad D \leq \frac{1-\rho}{1+\rho}\sigma^2, \tag{6}$$

and, similar to (4) and (5), we can develop lower and upper bounds for the case where X is a Gauss-Markov source [35]. Note that for $D > \frac{1-\rho}{1+\rho}\sigma^2$ the rate distortion function can be evaluated numerically [34]. For that range, which corresponds to $R < \log_2(1+\rho)$, distortion is always less than what is

[5]In general, for continuous memoryless sources under the MSE distortion constraint, Zamir [33] proves that $R_{X|Y}^{\mathrm{WZ}}(D) - R_{X|Y}(D) \leq \frac{1}{2}$ bits. Yet, such a bound is not know for the sources with memory.

[6]For target distortion $D > \sigma_0^2$ we do not need to transmit in the absence of impulse and $\underline{\mathrm{R}}_{X|Y}^{\mathrm{GBG}} = p_1 R_{X|Y,s_1}(D_i)$ in which $D_i = p_0\sigma_0^2 + p_1 D$.

evaluated by (6). Obviously, $\rho = 0$ gives a Gaussian source and (6) becomes identical to its Gaussian counterpart.

III. BCH-DFT Codes: Construction and Decoding

In this section, we study a class of real-number codes that are employed for binning throughout this paper, investigate some properties of their syndrome, and adapt their decoding algorithm to the Slepian-Wolf coding setup. These codes are a family of BCH codes in the real field whose parity-check matrix $\boldsymbol{H}$ and Generator matrix $\boldsymbol{G}$ are defined based on the DFT matrix; they are known as BCH-DFT codes, or simply DFT codes.

BCH-DFT codes [14] are linear block codes over the *real* or *complex* fields. Similar to other BCH codes, the spectrum of any codeword is zero in a block of $d \triangleq n-k$ cyclically adjacent components, where $d+1$ is the designed distance of the code [19]. The error correction capability of the code is, hence, given by $t = \lfloor \frac{d}{2} \rfloor$.

A. Encoding

The generator matrix of an (n,k) real BCH-DFT code [14] is defined by

$$\boldsymbol{G} = \sqrt{\frac{n}{k}} \boldsymbol{W}_n^H \boldsymbol{\Sigma} \boldsymbol{W}_k, \tag{7}$$

in which $\boldsymbol{W}_k$ and $\boldsymbol{W}_n^H$ respectively are the DFT and IDFT matrices of size k and n, and $\boldsymbol{\Sigma}$ is an $n \times k$ matrix defined as

$$\boldsymbol{\Sigma} = \begin{pmatrix} \boldsymbol{I}_\alpha & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{I}_\beta \end{pmatrix}, \tag{8}$$

where $\alpha = \lceil \frac{n}{2} \rceil - \lfloor \frac{n-k}{2} \rfloor$[7], $\beta = k - \alpha$, and the sizes of zero blocks are such that Σ is an $n \times k$ matrix [20]–[22], [36]. Then, for any $\boldsymbol{u}$, this enforces the spectrum of the codeword

$$\boldsymbol{c} = \boldsymbol{G}\boldsymbol{u}, \tag{9}$$

to have $n-k$ consecutive zeros, which is required for any BCH code [19]. The parity-check matrix $\boldsymbol{H}$, on the other hand, is constructed by using the $n-k$ columns of $\boldsymbol{W}_n^H$ corresponding to the $n-k$ zero rows of $\boldsymbol{\Sigma}$. Therefore, by virtue of the unitary property of $\boldsymbol{W}_n^H$, $\boldsymbol{H}$ is the null space of $\boldsymbol{G}$, i.e.,

$$\boldsymbol{H}\boldsymbol{G} = \boldsymbol{0}. \tag{10}$$

In the rest of this paper, we use the term "DFT code" in lieu of "real BCH-DFT code."

B. Decoding

Before introducing the decoding algorithm, we define some notation and basic concepts. Let $\boldsymbol{r} = \boldsymbol{c} + \boldsymbol{e}$ be the received vector (a noisy version of $\boldsymbol{c}$), where $\boldsymbol{c}$ is a codeword generated by (9). Suppose that $\boldsymbol{e}$ is an error vector with ν nonzero elements at positions $i_1, \ldots, i_\nu$; the magnitude of error at position i_p is e_{i_p}. Then, we can compute

$$\boldsymbol{s} = \boldsymbol{H}\boldsymbol{r} = \boldsymbol{H}(\boldsymbol{c}+\boldsymbol{e}) = \boldsymbol{H}\boldsymbol{e}, \tag{11}$$

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

$$s_m = \frac{1}{\sqrt{n}} \sum_{p=1}^{\nu} e_{i_p} X_p^{\alpha-1+m}, \qquad m = 1, \ldots, d \tag{12}$$

in which $\alpha = \lceil \frac{k+1}{2} \rceil$ as defined in (8), $X_p = e^{\frac{-j2\pi i_p}{n}}$, and $p = 1, \ldots, \nu$. Next, we define the syndrome matrix

$$\boldsymbol{S}_m = \begin{bmatrix} s_1 & s_2 & \cdots & s_{d-m+1} \\ s_2 & s_3 & \cdots & s_{d-m+2} \\ \vdots & \vdots & \ddots & \vdots \\ s_m & s_{m+1} & \cdots & s_d \end{bmatrix}, \tag{13}$$

for $\nu+1 \le m \le d-\nu+1$ [20]. Also, we define the covariance matrix as

$$\boldsymbol{R} = \boldsymbol{S}_m \boldsymbol{S}_m^H. \tag{14}$$

For decoding, the extension of the well-known Peterson-Gorenstein-Zierler (PGZ) algorithm to the real field [19] can be used. This *coding-theoretic* decoding, aimed at detecting, localizing, and calculating the errors, works based on the syndrome of error. We summarize the main steps of this algorithm, adapted for a DFT code of length n, in the following.

- **Error detection:** Determine the number of errors ν by constructing a syndrome matrix and finding its rank.
- **Error localization:** Find the coefficients $\Lambda_1, \ldots, \Lambda_\nu$ of the *error-locating polynomial* $\Lambda(x) = \prod_{i=1}^{\nu}(1 - xX_i^{-1})$ whose roots $X_1, \ldots, X_\nu$ are used to determine error locations; the errors are then in the locations $i_1, \ldots, i_\nu$ such that $X_1 = \omega^{i_1}, \ldots, X_\nu = \omega^{i_\nu}$ and $\omega = e^{-j\frac{2\pi}{n}}$.
- **Error calculation:** Finally, calculate the error magnitudes by solving a set of linear equations whose constants coefficients are powers of X_i.

In practice however, the received vector is distorted because of quantization. Let $\hat{\boldsymbol{c}}$ and $\boldsymbol{q}$ denote the quantized codeword and quantization noise so that $\hat{\boldsymbol{c}} = \boldsymbol{c} + \boldsymbol{q}$. Therefore, $\boldsymbol{r} = \hat{\boldsymbol{c}} + \boldsymbol{e}$ and its syndrome is no longer equal to the syndrome of error because

$$\boldsymbol{H}\boldsymbol{r} = \boldsymbol{H}(\boldsymbol{c}+\boldsymbol{q}+\boldsymbol{e}) = \boldsymbol{s}_q + \boldsymbol{s} = \tilde{\boldsymbol{s}}, \tag{15}$$

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

$$\tilde{s}_m = \frac{1}{\sqrt{n}} \sum_{p=1}^{\nu} e_{i_p} X_p^{\alpha-1+m} + \frac{1}{\sqrt{n}} \sum_{p'=1}^{n} q_{i_{p'}} X_{p'}^{p'-1}. \tag{16}$$

The distorted syndrome matrix $\tilde{\boldsymbol{S}}_m$ and the corresponding covariance matrix $\tilde{\boldsymbol{R}} = \tilde{\boldsymbol{S}}_m \tilde{\boldsymbol{S}}_m^H$ are defined similar to (13) and (14) but for the distorted syndrome samples.

While the exact value of the error is determined neglecting quantization, the decoding becomes an estimation problem in the presence of quantization. Then, it is imperative to modify the PGZ algorithm to decode the errors reliably [19]–[23]. An alternative approach is to use the *subspace-based* decoding. In the remainder of this section, we discuss this method and also improve the error detection and localization, by introducing a slightly different version of the existing methods.

[7] Knowing that n and k cannot be simultaneously even for a real DFT code [14], one can show that $\alpha = \lceil \frac{k+1}{2} \rceil$.

C. Modified Subspace-Based Decoding

The subspace-based error localization outperforms the coding theoretic one [20], [26]; it can be integrated into DSC in a straightforward fashion [37] once the syndrome of error is found. In addition to doing that, in this section, we introduce a new method for error detection.

1) Error detection: For a given DFT code, we first fix an empirical threshold θ based on eigendecomposition of $\tilde{\boldsymbol{R}}$ when the codewords are error-free, i.e. when only the quantization error exist. This threshold is on the magnitude of eigenvalues, rather than the determinant of $\tilde{\boldsymbol{R}}$. Let $\lambda_{\max}$ denote the largest eigenvalue of $\tilde{\boldsymbol{R}}$ for $m = t+1$. We find θ such that, for a desired probability of correct detection p_d,

$$\Pr(\lambda_{\max} < \theta) \geq p_d. \tag{17}$$

Note that $\lambda_{\max}$ is a random variable and we need to estimate its probability distributed function (pdf) for this purpose. In practice, when errors can occur, we estimate the number of errors by the number of eigenvalues of $\tilde{\boldsymbol{R}}$ greater than θ, as illustrated in Section VII and [37]. This one step estimation is better than the original estimation in the PGZ algorithm [19], [22], where the last row and column of $\boldsymbol{S}_t$ are removed until we come up with a non-singular matrix. The improvement comes from incorporating all syndrome samples, rather than some of them, for the decision making.

Ideally, we should set different thresholds depending on channel error powers; however, we choose one θ for all ranges to make the decoder simpler. Numerical results with different codes show that one suitable θ can be used for a wide range of error powers at the expense of a slight MSE performance degradation. When we bring down the threshold θ, effectively we let the decoder detect more errors. On the contrary, if we increase θ we allow less error detection. This implies a lower rate but may cause a higher MSE, as some of the previously-detected errors will be overlooked.

2) Error localization: The *subspace* or *coding-theoretic* error localizations can be used to find the coefficients $\Lambda_1, \ldots, \Lambda_\nu$ of the error-locating polynomial [20]. The subspace approach is, however, more general than the coding-theoretic approach in the sense that it can use up to $t+1-\nu$ degrees of freedom to localize ν errors, compared to just one degree of freedom in the coding-theoretic approach. This is because, the eigen-decomposition of the covariance matrix $\tilde{\boldsymbol{R}} = \tilde{\boldsymbol{S}}_m \tilde{\boldsymbol{S}}_m^H$ results in two orthogonal subspaces, namely the *error* and *noise* subspaces. There are $m-\nu$ vectors in the noise subspace; these are used to localize errors leveraging a line spectral estimation method, e.g., the multiple signal classification (MUSIC) algorithm [38]. Effectively, each vector corresponds to one error-locating polynomial [26], and by using the MUSIC approach we are averaging these $m-\nu$ polynomials to reduce the effect of quantization noise. Hence, in the subspace method, we get a better error localization compared with the coding-theoretic approach which is solely based on one polynomial. The best result is then obtained for $m = t+1$ [20] for which the size of $\tilde{\boldsymbol{S}}_m$ is either $(t+1)\times(t+1)$ or $(t+1)\times t$.

We apply the subspace-based error localization both to the syndrome- and parity-based DSC, similar to that in channel coding. However, one should note that in the DSC the syndrome of error is computed in a different manner from that in channel coding; this will be elaborated in Section IV.

3) Error calculation: This last step is rather simple. Let $\boldsymbol{H}_e$ be the matrix consisting of the columns of $\boldsymbol{H}$ corresponding to error indices. The errors magnitude $\boldsymbol{e} = [e_{i_1}, e_{i_2}, \ldots, e_{i_\nu}]^T$ can be determined by solving

$$\boldsymbol{H}_e \boldsymbol{e} = \tilde{\boldsymbol{s}}, \tag{18}$$

in a *least squares* sense, for example. This completes the error correction algorithm by calculating the error vector.

D. Performance Compared to Binary Codes

DFT codes by construction are capable of decreasing the quantization error. When there is no error, an (n,k) DFT code brings down the MSE below the quantization error level with a factor of $R_c = k/n$ [24], [25]. This is also shown to be valid for channel errors [23], as long as the channel can be modeled as an additive noise. To appreciate this, one can consider the generator matrix of a DFT code as *analysis frame operator* of a tight frame [25]; it is known that frames are resilient to any additive noise, and tight frames reduce the MSE k/n times [39]. Hence, DFT codes can result in a MSE even less than the quantization error level whereas the MSE in a binary code is obviously lower-bounded by the quantization error level.

IV. Wyner-Ziv Coding Using DFT Codes

In this section, we use DFT codes to do Wyner-Ziv coding in the real field. This is accomplished by using DFT codes for binning and transmitting the compressed signal, in the form of *syndrome* or *parity* samples, in a digital communication system. Let $\boldsymbol{x}$ be a sequence of real random variables $x_1 x_2 \ldots x_n$, and $\boldsymbol{y}$ be a noisy version of $\boldsymbol{x}$ such that $y_i = x_i + e_i$, where e_i is continuous, i.i.d., and independent of x_i, as described in (2). The lower-case letters x, y, and e, respectively, are used to show the realization of the random variables X, Y, and E. Since $\boldsymbol{e}$ is continuous, this model precisely captures any variation of $\boldsymbol{x}$, so it can be used to model the dependency between $\boldsymbol{x}$ and $\boldsymbol{y}$ accurately. This correlation model is important, for example, in video coders that exploit Wyner-Ziv concepts, e.g., when the decoder builds side information via extrapolation of previously decoded frames or interpolation of key frames [40].

In this paper, we study the GBG (and GE) correlation model and, for the sake of analysis, we assume that $\boldsymbol{e}$ contains up to t spikes (big errors) in each codeword.[8] For simplicity of presentation, the non-spiked errors are assumed to be zero in this section; this is as if we were using the GE model. However, when doing simulation for the GBG model, we use (2) with $p_0 + p_1 = 1$ and a reasonably small p_1 so that, for $\sigma_i^2 \gg \sigma_0^2$, with high probability there are t or less spikes in each codeword. Obviously, the remaining elements of $\boldsymbol{e}$ are non-zero for the GBG model.

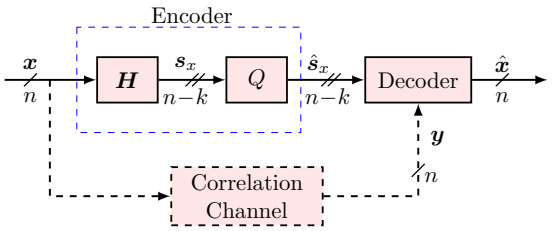

Fig. 2. The Wyner-Ziv coding using DFT codes: Syndrome approach.

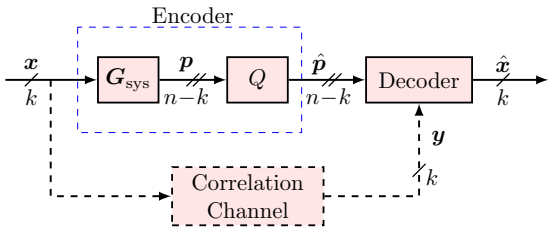

Fig. 3. The Wyner-Ziv coding using DFT codes: Parity approach.

A. Syndrome Approach

1) Encoding: Given $\boldsymbol{H}$, to compress an arbitrary sequence of data samples, we multiply it with $\boldsymbol{H}$ to find the corresponding syndrome samples $\boldsymbol{s}_x = \boldsymbol{H}\boldsymbol{x}$. The syndrome is then quantized ($\hat{\boldsymbol{s}}_x = \boldsymbol{s}_x + \boldsymbol{q}$) and transmitted over a noiseless digital communication system, as shown in Fig. 2. Note that $\boldsymbol{s}_x$, $\hat{\boldsymbol{s}}_x$ are both complex vectors of length $n-k$. Thus, it seems that to transmit each sample we need to send two real numbers, one for the real part and one for the imaginary part, which halves the compression ratio. However, we observe that the syndrome of a DFT code is symmetric, as stated below.

Lemma 1. *The syndrome of an (n,k) DFT code satisfies*

$$s_m = \begin{cases} s^*_{d-m+1}, & \text{if } k \text{ is odd,} \\ s^*_{d-m}, & \text{if } k \text{ is even,} \end{cases} \tag{19}$$

for $m = 1, \ldots, d$ and $d \triangleq n-k$.

Proof: The proof is straightforward; we show this for odd k and leave the other case to the reader. Since $\alpha = \lceil \frac{k+1}{2} \rceil$ and $d = n-k$, using (12), for odd k we can write

$$\begin{aligned} s_{d-m+1} &= \frac{1}{\sqrt{n}} \sum_{p=1}^{\nu} e_{i_p} X_p^{\frac{k-1}{2}+n-k-m+1} \\ &= \frac{1}{\sqrt{n}} \sum_{p=1}^{\nu} e_{i_p} X_p^{\frac{-k+1}{2}-m} = s^*_m. \end{aligned} \tag{20}$$

Note that, for any p, $X_p^n = 1$. ∎

The above lemma implies that, for any d, it suffices to know the first $\lceil \frac{d}{2} \rceil$ syndrome samples. We know that syndromes are complex numbers in general; however, from $s_{d-m+1} = s^*_m$ it is clear that if d is an odd number, s_m is real for $m = \lceil \frac{d}{2} \rceil$. Therefore, for an (n,k) code with odd k, transmitting $n-k$ real numbers suffices. This results in a compression ratio of $\eta_s = \frac{n}{n-k}$ for the binning step and thus for the encoder. Yet, one can check that for even k, we have to transmit $n-k+1$ real samples, which incurs a slight loss in compression. It is however negligible for large n.

2) Decoding: The decoder estimates the input sequence from the received syndrome and side information $\boldsymbol{y}$. To this end, it needs to evaluate the syndrome of (correlation) channel errors. This can be simply done by subtracting the received syndrome from the syndrome of the side information. Then, neglecting the quantization error, we obtain,

$$\boldsymbol{s}_e = \boldsymbol{s}_y - \boldsymbol{s}_x, \tag{21}$$

and $\boldsymbol{s}_e$ can be used to precisely estimate the error vector, as described in Section III-B. In practice, however, the decoder knows $\hat{\boldsymbol{s}}_x = \boldsymbol{s}_x + \boldsymbol{q}$ rather than $\boldsymbol{s}_{\boldsymbol{x}}$. Therefore, only a distorted syndrome of error is available, i.e.,

$$\tilde{\boldsymbol{s}}_e = \boldsymbol{s}_y - \hat{\boldsymbol{s}}_x = \boldsymbol{s}_e - \boldsymbol{q}. \tag{22}$$

Hence, using the PGZ algorithm, error correction is accomplished based on (22). Note that, having computed the syndrome of error, decoding algorithm in a DSC using DFT codes is exactly the same as that in the channel coding problem. This is different from DSC techniques in the binary field which usually require a slight modification in the corresponding channel coding algorithm to be customized for DSC.

B. Parity Approach

The syndrome-based Wyner-Ziv coding is straightforward, but it is not clear how we can use it for noisy transmission. In the sequel, we explore an alternative approached, namely parity-based approach, to the Wyner-Ziv coding.

1) Encoding: To compress $\boldsymbol{x}$, the encoder generates the corresponding parity sequence $\boldsymbol{p}$ with $n-k$ samples. The parity is then quantized and transmitted, as shown in Fig. 3, instead of transmitting the input data. To this end, we need to find a systematic generator matrix $\boldsymbol{G}_{\text{sys}}$, as $\boldsymbol{G}$ in (7) is not in the systematic form.

A first approach is to find $\boldsymbol{H}_{\text{sys}}$ and build $\boldsymbol{G}_{\text{sys}}$ based on that [41]. Another, simpler, way is to obtain a systematic generator matrix directly from $\boldsymbol{G}$. Let $\boldsymbol{G}_k$ be a square matrix of size k composed of arbitrary rows of $\boldsymbol{G}$. We see that $\boldsymbol{G}_k$ is invertible because using (7) any $k \times k$ submatrix of $\boldsymbol{G}$ can be represented as product of a Vandermonde matrix and the DFT matrix $\boldsymbol{W}_k$. This is also proven using a different approach in [25], where it is shown that any subframe of $\boldsymbol{G}$ is a frame, and its rank is equal to k. Hence, a systematic generator matrix is given by

$$\boldsymbol{G}_{\text{sys}} = \boldsymbol{G}\boldsymbol{G}_k^{-1}. \tag{23}$$

Besides, from $\boldsymbol{H}\boldsymbol{G} = \boldsymbol{0}$, it is clear that $\boldsymbol{H}\boldsymbol{G}_{\text{sys}} = \boldsymbol{0}$. Hence, we do not need to calculate $\boldsymbol{H}_{\text{sys}}$, and the same parity-check matrix $\boldsymbol{H}$ can be used for decoding in the parity approach. It is also obvious that $\boldsymbol{G}_{\text{sys}}$ is a real matrix. The question that remains to be answered is whether $\boldsymbol{G}_{\text{sys}}$ corresponds to a BCH code. To generate a BCH code, $\boldsymbol{G}_{\text{sys}}$ must have $n-k$ consecutive zeros in the transform domain. The Fourier transform of this matrix $\boldsymbol{W}_n\boldsymbol{G}_{\text{sys}} = (\boldsymbol{W}_n\boldsymbol{G})\boldsymbol{G}_1^{-1}$ satisfies the required condition because $\boldsymbol{W}_n\boldsymbol{G}$, the Fourier transform of original matrix, satisfies that.

[8] When doing simulation, to make the assumptions more realistic, we drop the constraint that a codeword cannot have more that t spikes.

It should be emphasized that one can arbitrarily choose the rows of $\boldsymbol{G}_k$ in (23); this gives $\binom{n}{k}$ systematic matrices for each (n,k) DFT code. Although any of the corresponding systematic codes can be used for encoding, the dynamic range of parity samples depends on their relative position of the chosen rows [36]. In [36, Theorem 7], we have proved that when using these systematic frames for error correction, the mean-squared reconstruction error is minimized when the systematic rows are chosen as evenly spaced as possible. In the extreme scenario, where the systematic rows are equally spaced, the systematic frame is also *tight*. This is realized only when n is an integer multiple of k. Such a frame lends itself well to minimize reconstruction error [24], [25], [42], [43]. Since the parity samples are error-free, the optimal $\boldsymbol{G}_{\text{sys}}$ reduces the occurrence of successive errors in one codeword. A similar idea is used in [44] to inhibits closely spaced sinusoids in oversampled DFT frames to increase spectral compression.

Finally, seeing that parity samples are real numbers, using an (n,k) DFT code, a compression ratio of $\eta_p = \frac{k}{n-k}$ is achieved. Obviously, a compression ratio of $\frac{n}{n-k}$ is achievable if we use a $(2n-k,n)$ DFT code.

2) Decoding: A parity decoder estimates the input sequence from the received parity and side information $\boldsymbol{y}$. Similar to the syndrome approach, at the decoder, we need to find the syndrome of (correlation) channel errors. To do so, we append the parity to the side information and form a vector of length n whose syndrome, neglecting quantization, is equal to the syndrome of error. That is,

$$\boldsymbol{z} = \begin{bmatrix} \boldsymbol{y} \\ \boldsymbol{p} \end{bmatrix} = \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{p} \end{bmatrix} + \begin{bmatrix} \boldsymbol{e} \\ \boldsymbol{0} \end{bmatrix} = \boldsymbol{G}_{\text{sys}}\boldsymbol{x} + \boldsymbol{e}', \tag{24}$$

and $\boldsymbol{e}' = [\boldsymbol{e} \mid \boldsymbol{0}]^T$. Hence,

$$\boldsymbol{s}_z = \boldsymbol{s}_{e'}. \tag{25}$$

Similarly, when quantization is involved ($\hat{\boldsymbol{p}} = \boldsymbol{p} + \boldsymbol{q}$), we get

$$\tilde{\boldsymbol{z}} = \begin{bmatrix} \boldsymbol{y} \\ \hat{\boldsymbol{p}} \end{bmatrix} = \boldsymbol{z} + \begin{bmatrix} \boldsymbol{0} \\ \boldsymbol{q} \end{bmatrix} = \boldsymbol{G}_{\text{sys}}\boldsymbol{x} + \boldsymbol{e}' + \boldsymbol{q}', \tag{26}$$

and $\boldsymbol{s}_{\tilde{z}} = \boldsymbol{s}_{e'} + \boldsymbol{s}_{q'}$, where, $\boldsymbol{q}' = [\boldsymbol{q} \mid \boldsymbol{0}]^T$, and $\boldsymbol{s}_{q'} \equiv \boldsymbol{H}\boldsymbol{q}'$. Therefore, we obtain a distorted version of error syndrome. In both cases, the rest of the algorithm, which is based on the syndrome of error, is similar to that in the channel coding problem using DFT codes, as explained in Section III-C2.

Error localization algorithm for the parity-based DSC [41] can be further improved using the fact that parity samples are error-free. As parity samples are transmitted over a noiseless channel, the error locations, in the codewords, are restricted to the systematic samples. Therefore, we can exclude the set of roots corresponding to the location of the parity samples. We call this *adapted* error localization. Furthermore, it makes sense to use a code with evenly-spaced parity samples so as to optimize the location of error-free and error-prone samples in the codewords. Such a code maximizes the distance between the error-prone roots of the code; hence, it helps decrease the probability of incorrect decision.

C. Comparison Between the Two Approaches

1) Rate: As it was shown earlier, using an (n,k) code the compression ratio in the syndrome and parity approaches is $\eta_s = \frac{n}{n-k}$ and $\eta_p = \frac{k}{n-k}$, respectively. Hence, for a given code, the parity approach is $\frac{k}{n} < 1$ times less efficient than the syndrome approach. Conversely, we can find two different codes that result in same compression ratio η, say $\frac{n}{n-k}$. We mentioned that in the parity approach, a $(2n-k,n)$ code can be used for this matter, whereas an (n,k) DFT code gives the desired compression ratio in the syndrome approach. Thus, for a given compression ratio the syndrome approach implies a code with smaller rate compared to the code required in the parity approach.

2) Delay: Assuming the delay imposed by a system to decode each block of code depends on the number of samples to be transmitted, the delay in the parity approach is larger than in the syndrome approach. More precisely, for the compression ratio of $\eta = \frac{n}{n-k}$, the delay in the former approach is proportional to $n-k$ while the it is proportional to n in the latter approach. These are the length of syndrome and parity vectors, respectively.

3) Performance: From frame theory, we know that DFT frames are tight, and an (n,k) tight frame reduces the quantization error with a factor of $R_c = \frac{k}{n}$ [24], [25], [39]. This result is extended to errors, given that channel can be modeled by an additive noise [23]. The MSE performance of systematic DFT frames also linearly depends on the code rate, though they are not necessarily tight [36]. Therefore, for codes with the same error correction capability, the lower the code rate the better the error correction performance. This implies a better performance for syndrome-based DSC. Further, a $(2n-k,n)$ code has $n-k$ roots more than an (n,k) code on the unit circle; hence, the roots are closer to each other and the probability of incorrect localization of errors increases.

Additionally, from rate-distortion theory we know that the rate required to transmit a Gaussian source logarithmically increases with the source variance [45]. Thus, in a system that uses a real-number code for encoding, since coding is performed before quantization, the variance of transmitted sequence depends on the behavior of the encoding matrix. In the syndrome-based DSC we transmit $\boldsymbol{s} = \boldsymbol{H}\boldsymbol{x}$. One can check that $\sigma_{\boldsymbol{s}} = \sigma_{\boldsymbol{x}}$ [36]. Unlike that, in the parity-based DSC, the variance of the parity samples is larger than that of the inputs. More precisely, in an (n,k) systematic DFT code, if $\boldsymbol{c} = \boldsymbol{G}_{\text{sys}}\boldsymbol{x}$, then $\sigma_{\boldsymbol{c}}^2 = \gamma\sigma_{\boldsymbol{x}}^2$ where $\gamma = \frac{1}{n}\operatorname{tr}\left(\boldsymbol{G}_{\text{sys}}^H\boldsymbol{G}_{\text{sys}}\right) \geq 1$ [36]. Since we can write $\boldsymbol{c} = [\boldsymbol{x} \mid \boldsymbol{p}]^T$, we have

$$\sigma_{\boldsymbol{p}}^2 = \frac{\gamma n - k}{n-k}\sigma_{\boldsymbol{x}}^2 \geq \sigma_{\boldsymbol{x}}^2. \tag{27}$$

From [36, Theorem 7], we know that the smallest $\sigma_{\boldsymbol{p}}$ for a given DFT code is achieved when the parity samples, in the corresponding codewords, are located as "evenly" as possible. Furthermore, the equality in (27) can be achieved only when $n = Mk$ where M is an integer greater than or equal to 2 [36]. Thus, such a code cannot be used in the parity-based DSC as $M \geq 2\,(n \geq 2k)$ result in signal "expansion" rather than compression. Obviously though, we can use such a code when

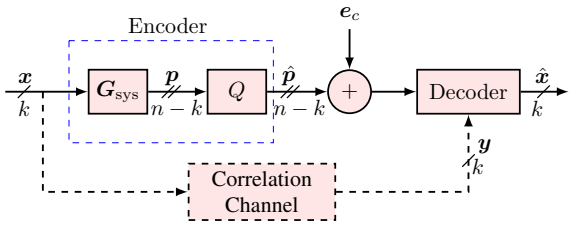

Fig. 4. Joint source-channel coding (JSCC) with side information at the decoder based on DFT codes. This scheme can be straightforwardly extended to distributed JSCC.

expansion is allowed, for example in joint source-channel coding, as we will see the next section.

Considering the above arguments, it is reasonable to expect the syndrome-based DSC perform better than the parity-based one, for a given code or a fixed compression ratio. This is verified numerically in Section VII. The parity-based DSC, however, has other advantages. For example, it can be easily extended to distributed joint source-channel coding, as explained in the following section.

V. Distributed Joint Source and Channel Coding

The concept of lossy DSC and Wyner-Ziv coding using DFT codes was explained both for the syndrome and parity approaches in Section IV, where syndrome or parity samples are quantized and transmitted over a *noiseless* channel. This implies *separate* source and channel coding. Although simple, the *separation theorem* is based on several assumptions, such as the source and channel coders not being constrained in terms of complexity and delay, which do not hold in many situations. It breaks down, for example, for non-ergodic channels and real-time communication. In such cases, it makes sense to integrate the design of the source and channel coder systems, because *joint source-channel coding* (JSCC) can perform better given a fixed complexity and/or delay constraints. Likewise, distributed JSCC (DJSCC) has been shown to outperform separate distributed source and channel coding in some practical cases [46]. DJSCC has been addressed in [9], [46]–[48], using different binary codes.

In this section, we extent the parity-based Wyner-Ziv coding of analog sources to the case where errors in the transmission can happen. Thus, we introduce distributed JSSC of analog correlated sources in the analog domain. To do this, we use a single DFT code both to *compress* $\boldsymbol{x}$ and *protect* it against channel variations; this gives rise to a new framework for DJSCC, in which quantization is performed after doing JSCC in the analog domain. This scheme directly maps short source blocks into channel blocks, and thus it is well suited to low-delay coding.

A. Coding and Compression

To compress and protect $\boldsymbol{x}$, the encoder generates the parity sequence $\boldsymbol{p}$ of $n-k$ samples, with respect to a good systematic DFT code. The parity is then quantized and transmitted over a noisy channel, as shown in Fig. 4. To keep the dynamic range of parity samples as small as possible, we make use of optimal systematic DFT codes, proposed in [36]. This increases the efficiency of the system for a fixed number of quantization levels. Using an (n,k) DFT code a total compression ratio of $\eta = \frac{k}{n-k}$ is achieved. Obviously, if $n < 2k$ compression is possible ($\eta > 1$). However, since there is little redundancy the end-to-end distortion could be high. Conversely, a code with $n > 2k$ ($\eta < 1$) expands input sequence by adding *soft redundancy* to protect it in a noisy channel.

B. Decoding

Let $\tilde{\boldsymbol{p}} = \hat{\boldsymbol{p}} + \boldsymbol{e}_c$ be the received parity vector which is distorted by quantization error $\boldsymbol{q}$ ($\hat{\boldsymbol{p}} = \boldsymbol{p} + \boldsymbol{q}$) and channel error $\boldsymbol{e}_c$. Also, let $\boldsymbol{y} = \boldsymbol{x} + \boldsymbol{e}_v$ denote side information where $\boldsymbol{e}_v$ represents the error due to the "virtual" correlation channel. The objective of the decoder is to estimate the input sequence from the received parity and side information. Although we only need to determine $\boldsymbol{e}_v$, effectively it is required to find both $\boldsymbol{e}_v$ and $\boldsymbol{e}_c$. From an error correction point of view, this is equal to finding the error vector $\boldsymbol{e} = [\boldsymbol{e}_v \;\; \boldsymbol{e}_c]^T$ that affects the codeword $[\boldsymbol{x} \;\; \boldsymbol{p}]^T$. Hence, to find the syndrome of error at the decoder, we append the parity $\tilde{\boldsymbol{p}}$ to the side information $\boldsymbol{y}$ and form $\tilde{\boldsymbol{z}}$, a valid codeword perturbed by quantization and channel errors,

$$\tilde{\boldsymbol{z}} = \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{p} \end{bmatrix} + \begin{bmatrix} \boldsymbol{e}_v \\ \boldsymbol{e}_c \end{bmatrix} + \begin{bmatrix} \boldsymbol{0} \\ \boldsymbol{q} \end{bmatrix} = \boldsymbol{G}_{\text{sys}}\boldsymbol{x} + \boldsymbol{e} + \boldsymbol{q}'. \quad (28)$$

Multiplying both sides by $\boldsymbol{H}$, we obtain

$$\boldsymbol{s}_{\tilde{z}} = \boldsymbol{s}_e + \boldsymbol{s}_{q'}, \quad (29)$$

where $\boldsymbol{s}_e \equiv \boldsymbol{H}\boldsymbol{e}$ and $\boldsymbol{s}_{q'} \equiv \boldsymbol{H}\boldsymbol{q}'$. Again, we use the GE model with $q_2 = 0$ in (2) to generate $\boldsymbol{e}$. It should be emphasized that for $\boldsymbol{q} = \boldsymbol{0}$, error vector can be determined exactly, as long as the number of errors is not greater than t. In practice, quantization is also involved, and we obtain only a distorted version of error syndrome. Knowing the syndrome of error, we use the error detection and localization algorithm, explained in Section III-B, to find and correct error.

VI. Rate-Adaptive DSC

The proposed DSC schemes, both the parity and syndrome approaches, are suitable for low-delay coding as DFT codes perform sufficiently well for short blocks. On the other hand, a short code is more vulnerable to the variations of channel. Expectedly, the performance of the proposed systems degrades when the correlation between the sources is unstable. 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; these systems are popular in the transmission of non-ergodic data, like video. Rate-adaptive DSC based on binary codes, e.g., puncturing the parity or syndrome bits of turbo and LDPC codes, have been proposed in [49], [50]. Although puncturing the parity or syndrome samples can be used for rate-adaption in our systems, it severely affects the decoding algorithm and substantially increases the MSE.

We propose an alternative, more efficient method to perform DSC in a rate-adaptive fashion. The algorithm, which works based on additional syndrome samples, is based on a straightforward extension of the subspace decoding, proposed by the

authors [26] to enhance the error localization of quantized DFT codes. Increasing the number of syndrome samples makes it possible to enlarges the dimension of the noise subspace and thus to increase the number of error-locating polynomials; the reader is referred to [26] for further details. Then, instead of using one polynomial for decoding, one can superimpose many polynomials to diminish the effect of quantization error and improve error localization to a large extent. All we need is to have some extra syndrome samples of error vector at the decoder. The proposed algorithm is naturally suitable for rate adaption in the syndrome-based DSC.[9] This is because the decoder is able to compute extra syndromes of error upon receiving some new syndrome samples of the data vector. To this end, the encoder and decoder are required to agree on another parity check matrix $\bar{\boldsymbol{H}}$, which we dub *extended parity check matrix* and form it based on those k columns of $\boldsymbol{W}_n^H$ (the IDFT matrix of order n) that are not used to build $\boldsymbol{H}$. Then similar to the syndrome vector $\boldsymbol{s}$, the extended syndrome vector $\bar{\boldsymbol{s}}$ is defined as

$$\bar{\boldsymbol{s}} = \bar{\boldsymbol{H}}\boldsymbol{r} = \bar{\boldsymbol{H}}\boldsymbol{c} + \bar{\boldsymbol{H}}\boldsymbol{e}. \tag{30}$$

But $\bar{\boldsymbol{H}}\boldsymbol{c}$ is not necessarily zero and should be compensated for. This can be done naturally, based on the following rate adaptation algorithm:

1) The decoder requests some extra syndrome samples via a feedback channel and based on the estimated number of errors, e.g. when $\hat{\nu} > t$
2) The encoder computes $\bar{\boldsymbol{s}}_x = \bar{\boldsymbol{H}}\boldsymbol{x}$ and transmits it to the decoder sample by sample
3) The decoder computes $\bar{\boldsymbol{s}}_y = \bar{\boldsymbol{H}}\boldsymbol{y} = \bar{\boldsymbol{s}}_x + \bar{\boldsymbol{s}}_e$ to find $\bar{\boldsymbol{s}}_e = \bar{\boldsymbol{s}}_y - \bar{\boldsymbol{s}}_x$ and append it to $\boldsymbol{s}_e$ in order to use the extended subspace decoding algorithm.

This algorithm works both in the syndrome- and parity-based DSC. In the syndrome-based DSC, the encoder transmits a short syndrome based on an (n, k) code and augments it with additional samples, if more samples are requested. The algorithm is incremental so that there is no need to re-encode the sources when extra syndrome samples are requested. Likewise, in the parity-based DSC if the received parity samples are not enough to decode a block, the decoder requests for extra syndrome samples. Note that the decoding algorithm, even in the parity-based DSC, is based on syndrome of error. That is, no matter which of parity or syndrome is transmitted to the decoder, the decoding algorithm needs to compute the syndrome of error.

In the above algorithm, the decision to request more syndrome samples is based on the estimated number of errors and the decoder does not need to decode the whole block. On the contrary, the decision to request extra syndrome samples in a rate-adaptive DSC systems based on LDPC codes, and any binary codes in general, is based on the decoding of the whole long block which is computationally less efficient and requires more time. Another importance of the rate adaptation is to find out whether or not the value of the error detection threshold θ is appropriate, especially if we have no estimation of the CEQNR. We know that by bringing the threshold down, we can increase the probability of error detection. On the contrary, if we increase the threshold fewer errors will be detected and thus rate adaptation is required to a lesser extent. With this in mind, if there are too many requests for extra syndromes we may increase θ; conversely, if there are not many requests for rate adaptation we can bring the threshold down as long as it improves the end-to-end distortion.

[9]It can, however, be applied to the the parity-based DSC in the same way.

VII. Simulation Results

We evaluate the performance of the proposed systems using a Gauss-Markov source $\boldsymbol{x}$ with mean zero, variance one, and correlation coefficient $0 \leq \rho < 1$ generated by

$$x_i = \sqrt{1-\rho^2}z_i + \rho x_{i-1}, \tag{31}$$

in which $\boldsymbol{z} = \{z_i\}$ is a zero-mean, unit-variance i.i.d. Gaussian process. The side-information is generated using the forward correlation model as presented in Section II-B. That is, for the GBG model there are both background noise and impulsive errors while for the GE model only the impulsive errors exist. The background noise, generated by $\mathcal{N}(0, \sigma_0^2)$, affects each and every sample of $\boldsymbol{x}$ while the impulsive error is added to a fraction of samples of $\boldsymbol{x}$. The amplitude of impulses is generated based on $\mathcal{N}(0, \sigma_e^2)$ and their position is also selected randomly. To compress the data, $\boldsymbol{x}$ is binned using a (n, k) DFT code. The compressed vector, either syndrome or parity, is then quantized with a b-bit uniform quantizer ($4 \leq b \leq 6$) and transmitted over a communication channel. The quantizer step size Δ depends on the effective range of the compressed data (i.e., the syndrome or parity) and thus it can be different for the syndrome and parity approaches. It also depends on the percent of the data we would like to be in the range of quantizer.[10] For an input range of $[-m\sigma, m\sigma]$ we get $\Delta = \frac{2m\sigma}{2^b}$. The decoder detects, localizes, and decodes errors. We compute the MSE between the transmitted and reconstructed data, to measurer end-to-end distortion. In all simulations, we use 10^6 input samples for each channel-error-to-quantization-noise ratio (CEQNR), defined as σ_e^2/σ_q^2 where $\sigma_q^2 = \frac{\Delta^2}{12}$. We vary the CEQNR from 10dB to 40dB and plot the resulting MSE. A larger CEQNR corresponds to a larger variation of the error amplitude than that of the quantization noise power. Numerical results are presented in two different subsections for syndrome-based DSC and its extension to rate-adaptive DSC, and parity-based DSC and its extension to DJSCC.

A. Syndrome-Based DSC and Rate-Adaptive DSC

Before showing the simulation results for the reconstruction distortion, we elaborate how to choose the parameter θ and the way it affects the MSE. As we explained in Section III-C1, θ is used to estimate the number of errors, and from (17) we need to have the pdf of $\lambda_{\max}$ to find θ for specific p_d. So, the first step is to evaluate the pdf of $\lambda_{\max}$; this is done based on the eigendecomposition of $\tilde{\boldsymbol{R}}$ for the quantized code where there is only quantization error. We then need to fix the value of p_d to effectively estimate the number of errors.

[10]For one thing, $[-4\sigma_x, 4\sigma_x]$ contains more than 99.99% of the input data, where σ_x^2 is the variance of x.

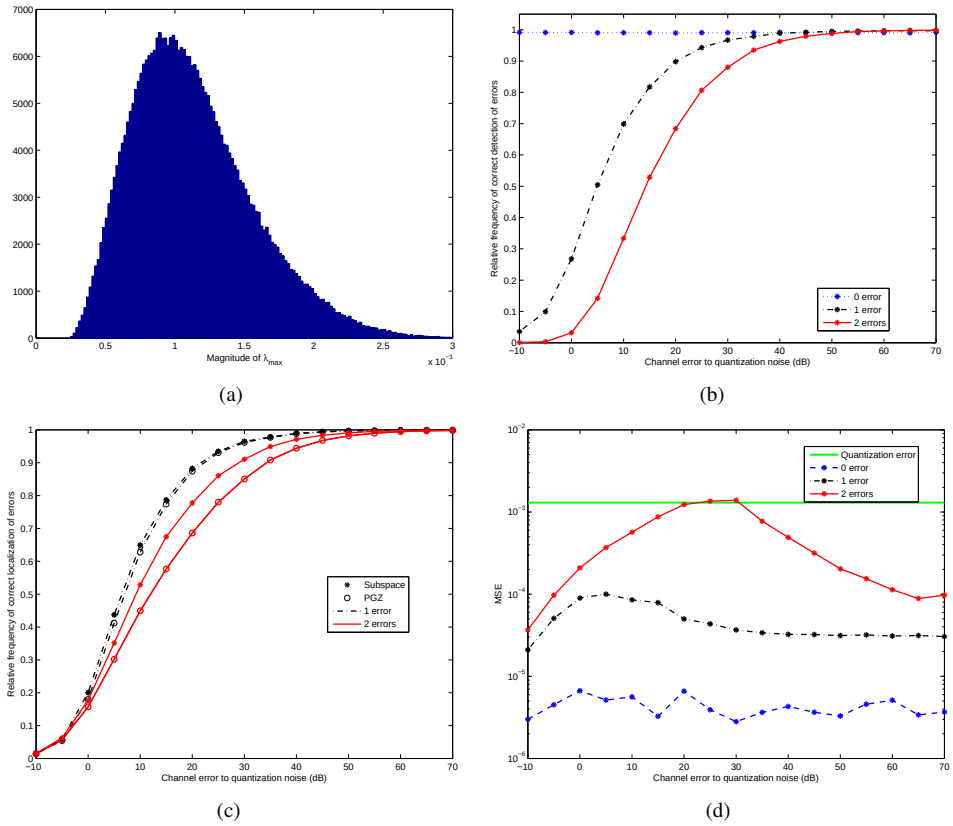

Fig. 5. Performance evaluation of the syndrome-based DSC proposed in Fig. 2, for a $(10, 5)$ DFT code and different number of errors in each block of the code. The results are based on a Gauss-Markov source with $\rho = 0.09$ and a quantizer with $m = 4$ and $b = 6$. (a) Histogram of $\lambda_{\max}(\tilde{\boldsymbol{R}})$ for the quantized code. This is used to set a threshold for detection. (b) Probability of correct detection of errors for $\theta = 0.0064$. (c) Probability of correct localization of errors based on Fig. 5(b). (d) The end-to-end distortion for subspace-based error localization given in Fig. 5(c).

Numerical results show that $p_d = 90 - 99\%$ is a good initial value for θ. Increasing θ will decrease $\hat{\nu}$, the estimated number of errors, whereas decreasing θ will increase that. Meanwhile, by decreasing θ the probability of false detection increases. So the there is a trade off between correct detection and false alarm. By changing θ in reasonably small steps and evaluating the end-to-end distortion, one can find its optimal value so as to minimize the MSE.[11] The optimal value of θ varies based on the CEQNR, even though this variation is small for a small range of CEQNR. Despite that, in our simulations, for each code, we use one θ at all CEQNRs so that the decoder does not need to know the value of CEQNR. If CEQNR is know at the decoder, we can assign a more accurate θ to get a slightly better MSE.

At very low CEQNRs, although error localization is poor, the MSE is still very low because compared to the quantization error, the errors can be so small that the algorithm does not detect (and localize) them. Instead, it may occasionally localize and correct quantization errors. Note that, even if no errors are localized and corrected, the MSE is still very small as the errors are negligible at very low CEQNRs. Additionally, recall that the MSE is always reduced with a factor of $R_c = \frac{k}{n}$, in an (n, k) DFT code.

To familiarize the reader with the decoding steps, in Fig. 5, we detail the decoding steps for a $(10, 5)$ DFT code for the GE correlation model. First, based on Fig. 5(a), the threshold $\theta_0 = 0.0024$ is found for $p_d = 99\%$. Next, this is used to estimate ν in Fig. 5(b). The estimated ν is subsequently used to find the location of errors, both for the PGZ and subspace-based error localization, in Fig. 5(c). Then, the output of Fig. 5(c), for the subspace method, is fed to the last step to find the magnitude of errors and correct them. The resulting MSE is depicted in Fig. 5(d). In the remainder of this section, we will focus only on the MSE performance without plotting the results for the intermediate steps (i.e., error detection and localization). We should point out that $\theta_0 = 0.0024$ gives an initial value and needs to be optimized depending on the CEQNR. So we change θ_0 by a step of 0.001 and check the resulting MSE. Numerical results show that $\theta = 0.0064$, which corresponds to $pd = 0.9995$, result in minimum average MSE at CEQNR between 10dB to 40dB. Clearly, if we change the range of CEQNR the optimal θ will differ. However, its impact on the average MSE is very small.

We begin with the syndrome approach and probe its performance by varying the CEQNR for different codes with different compression ratios. We use the subspace-based method for error localization since its performance is better than the PGZ method. To put our results in perspective, we also calculate the MSE for the *genie-aided* error localization in which the true location of errors is given to the decoder. The genie-aided decoding reflects the ideal performance of the proposed DSC system, when least square decoding is used in the last step of

[11] Note that finding the optimal value of θ can be done off-line or using pilot data.

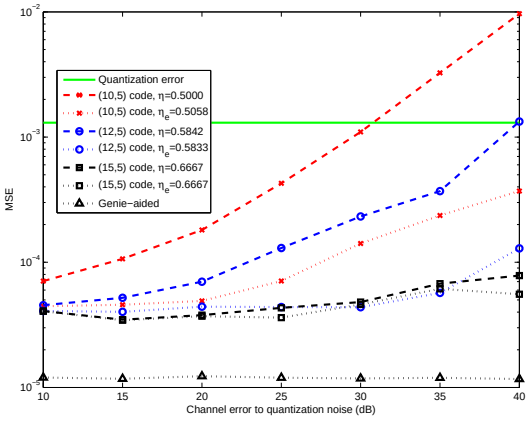

Fig. 6. The performance of rate-adaptive DSC versus non-adaptive approach for three different DFT codes with the GE correlation model with $p_1 = 0.04$, $\rho = 0.9$, a 6-bit quantizer, and $\theta = 0.0064$. The average rates (η_e) for the rate-adaptive approach are a little more than their original code rates (η) while the gain in the MSE is noticeably high. For the $(15, 5)$ code $\eta_e = 0.66669$ and $\eta = 0.66667$, thus there is not much gain from rate-adaption.

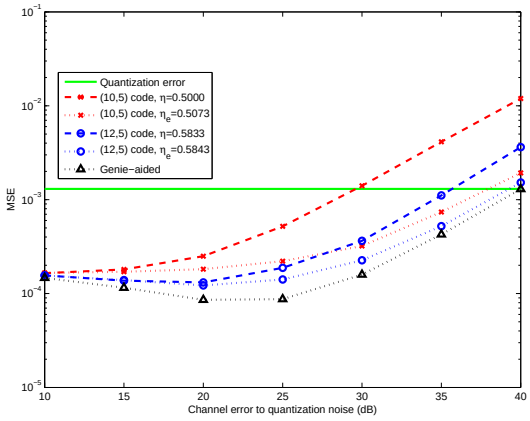

Fig. 7. The MSE performance of the rate-adaptive DSC versus non-adaptive approach for a Gauss-Markov source with $\rho = 0.9$ and the GBG correlation model with $\sigma_0 = 0.01\sigma_e, p_1 = 0.04$. The average rate increases due to the rate-adaption are still very small while reduction in the MSE is considerable.

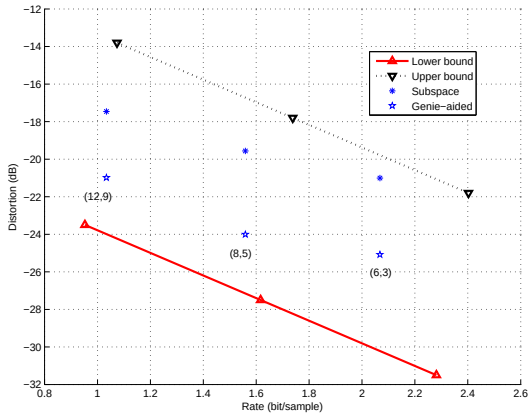

Fig. 8. The distortion-rate function and bounds for coding a Gauss-Markov source X with $\sigma_X = 1$, $\rho = 0.9$ quantized with a 4-bit quantizer. The graph is based on the GBG correlation model with $\sigma_0 = 0.05\sigma_e$ at CEQNR= 25dB, or equivalently $\sigma_0 = 0.1282$ and $\sigma_e = 2.5647$ for $b = 4$. The asterisks show the achievable points based on $(12, 9)$, $(8, 5)$, $(6, 3)$ codes, respectively, while the stars show the achievable points for the same codes assuming a perfect error localization. Rate-adaptation has been applied in both cases.

the decoding (i.e., the error calculation).

Figure 6 presents the MSE performance for three different codes, namely $(10, 5)$, $(12, 5)$ and $(15, 5)$ codes, without and with rate adaptation. Expectedly, as the code rate increases, which implies a lower compression ratio, the MSE goes down. For rate adaptation, the decoder requests extra syndrome samples if $\hat{\nu} \geq t$. Upon receiving the new syndrome samples, the decoder applies the extended subspace method for error localization and uses this new result for error correction. As it can bee seen from Fig. 6, rate adaptation can noticeably improve the performance at the expense of negligible increase in the effective code rate. This is particularly important when the codeword length n is short. With a small n, the probability of having more than t errors in one block can be relatively high. By requesting more syndrome samples and using the extended subspace decoding algorithm, a rate-adaptive system effectively increases the error correction capability of the code, and thus improves the overall performance. Another important use of rate adaptation is to compensate for the fixed threshold at different CEQNR. As explained earlier, ideally, we should have different thresholds at each CEQNR even though, for simplicity of decoding, we use one θ for all ranges of CEQNR. When a fixed threshold is used at different CEQNRs, effectively we let the decoder detect more errors at higher CEQNRs rather than the lower ones. Then, rate adaptation comes in handy when there are more than t errors in one block. Even if there are t errors in one block, one might use a few more syndrome samples to enhance the decoding performance. This improves the end-to-end distortion at the expense of increasing the rate.

We now evaluate the performance of the syndrome-based DSC for the GBG model in Fig. 7. It can be seen that rate adaptation decreases the MSE in both cases and the average rate increase due to that is still very small.

Finally, we plot the distortion rate region for the syndrome-based DSC and the GBG model and a 4-bit quantizer in Fig. 8. The goal is to compare the rate distortion performance of the system with the asymptotic bounds. Since the rate distortion function for the GBG model is not known we compare the results with the upper and lower bounds introduced in (4) and (5). It should be emphasized that the lower bound is based on the assumption that both encoder and decoder know the true position of the errors (impulses); this can make the lower bound very loose. When the encoder and/or decoder does not have this knowledge the lower bound is expected to be tighter; i.e., it will shift upward and get closer to the achievable points. To visualize the encouraging potential of the proposed framework, we have also shown the performance of our system for the genie-aided (perfect) error localization.

Seeing that we do not use the ideal Slepian-Wolf coding assumption ($n \to \infty$), the gap between performance of our schemes and theoretical rate-distortion function is expected to

be more than usual. However, it should be noted that capacity-approaching channel codes introduce significant delay if one strives to approach the capacity of the channel with a very low probability of transmission error. Hence, those are out of the question for delay-sensitive systems. In that case, it would be best to use channel codes of low rate and focus on achieving a very low probability of error. The system we introduced is a low-delay system which works well with reasonably high-rate codes. This is because the block length in our system is nb bits (n samples) and n is usually much smaller than the code length in LDPC and turbo codes. For example, in Fig. 8 all points are for $n \leq 12$ and $b = 4$ which results in block length $L \leq 48$ at the worst case, while it is common to have codes of length 10^5 in DSC based on LDPC codes.

In Fig. 8 we also envisage the performance of our system for the genie-aided (perfect) error localization. It indicates the great potential of the proposed framework to become much closer to the theoretical lower bound and encourages investigation on better error localization algorithms. Furthermore, we know that for sources with memory there is dependency between the samples and vector quantization (VQ) can exploit this dependency. The gain resulting from the memory advantage can be considerably high at high rates [34]; therefore, it is reasonable to expect that VQ would improve the rate distortion performance of our system.

B. Parity-based DSC and DJSCC

The second part of numerical results focuses on the parity-based DSC and its extension to the noisy channel setting, i.e., DJSCC. We do simulations both for a Gauss-Markov source with $\rho = 0.9$ and Gaussian sources, i.e., a Gauss-Markov source with $\rho = 0$. We first compare the performance of the parity- and syndrome- based approaches for two codes with the same compression ratio using a Gauss-Markov source with $\rho = 0.9$. We use a $(5, 1)$ code for the syndrome approach and a $(9, 5)$ code for the parity approach; thus, the compression ratio for both codes is $\eta = 0.8$. The results are presented in Fig. 9; it can be seen that syndrome-based DSC performs better than the parity-based DSC. As we explained in Section IV-C3, the performance of the parity-based system is not as good as that of the syndrome-based system since $\sigma_{\boldsymbol{p}} > \sigma_{\boldsymbol{s}}$, and this implies a that the rate required to transmit the parity samples is more than that of the syndrome samples given a same distortion.

We next plot the distortion-rate function for coding a Gauss-Markov source with $\rho = 0$ (Gaussian source) based on the parity-based DSC and compare its performance against the theoretical limits in Fig. 10. The points are based on $(19, 17)$, $(9, 7)$, $(8, 5)$ codes, respectively. As we mentioned earlier, the performance of the parity-based DSC depends on the position of the parity samples. To find the rate distortion pairs in the above figure, the position of parity samples were chosen as even as possible, to minimize the MSE.

Finally, we compare the performance of the proposed system for DJSCC using three codes $(5, 3)$, $(10, 5)$, and $(15, 5)$, with compression ratios equal to $\frac{2}{3}$, 1, and 2, respectively.

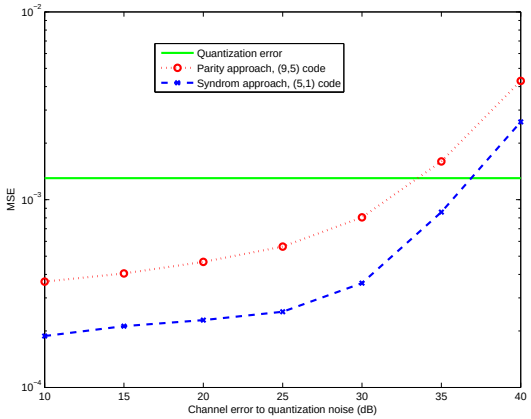

Fig. 9. The MSE performance of syndrome- parity-based DSC for a Gauss-Markov source with $\rho = 0.9$ and the GBG correlation model with $\sigma_0 = 0.01\sigma_e$, $p_1 = 0.04$. The compression ratio for both approaches is 0.8 as the corresponding codes are $(5, 1)$ and $(9, 5)$ for the syndrome and parity approaches, respectively.

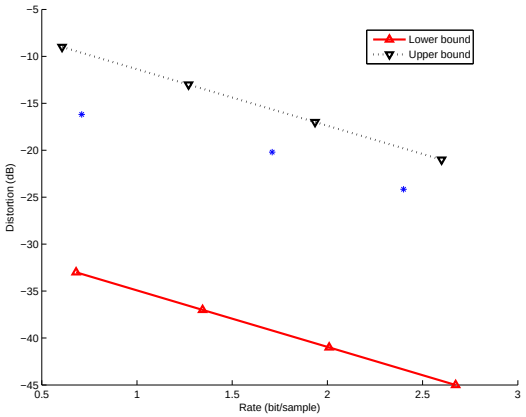

Fig. 10. The distortion-rate function and bounds for coding a Gaussian source X with $\sigma_X = 1$ and the GBG correlation model with $\sigma_0 = 0.05\sigma_e$ at CEQNR= 25dB and $b = 6$. The achievable points are based on $(19, 17)$, $(9, 7)$, $(8, 5)$ codes, respectively.

Again, for each code we use a $\boldsymbol{G}_{\text{sys}}$ with the best MSE performance, in light of [36, Theorem 7]. Specifically, rows $\{1, 3, 5\}$ are chosen as the systematic rows of $\boldsymbol{G}_{\text{sys}}$ for the $(5, 3)$. For the $(10, 5)$ and $(15, 5)$ codes the optimal solution is to choose, respectively, every second and third rows as the systematic rows. The remaining rows correspond to the parity samples. As expected, in Fig. 11 it can be seen that when code rate decreases the MSE decreases. Note that, in this simulation 3% of parity samples are affected by an impulsive noise whose power is the same as the power of impulses in the correlation channel. The $(10, 5)$ code neither compresses nor expands the input source as it transmit 5 parity samples in lieu of every 5 source samples; nevertheless, it can combat the transmission noise without adding any redundancy.

VIII. Conclusions

We have introduced a new framework for the distributed lossy source coding, in general, and the Wyner-Ziv coding, in particular. The idea is to do binning before quantizing

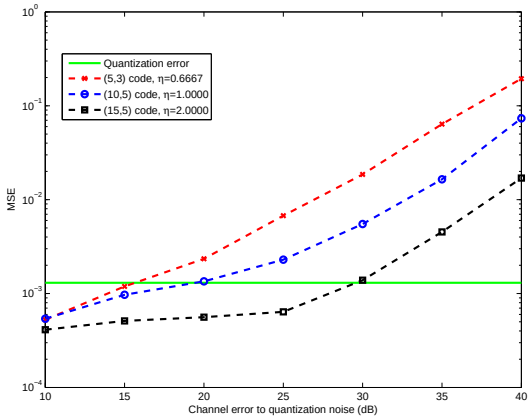

Fig. 11. The MSE performance of the DJSCC for the GBG correlation model with $\sigma_0 = 0.05\sigma_e, p_1 = 0.03$ and $\rho = 0$. The compression ratio for the codes are equal to $\frac{2}{3}$, 1, and 2, implying compression for the first code and expansion for the third code. For the $(10, 5)$ code there is neither compression nor expansion since $\eta = 1$.

the continuous-valued signal, as opposed to the conventional approach where binning is done after quantization. By doing binning in the real field, the virtual correlation channel can be modeled more accurately and the quantization error can be compensated for when there is no error. In the new paradigm, Wyner-Ziv coding is realized by cascading a Slepian-Wolf encoder with a quantizer. We employ BCH-DFT codes to do binning in the real field and we introduce both syndrome- and parity-based DSC systems. The extension of the parity-based Wyner-Ziv coding to joint source-channel coding with side information at the decoder is straightforward. This scheme directly maps short source blocks into channel blocks, and thus it is appropriate for low-delay coding. From simulation results, we conclude that our systems can improve the reconstruction error even using short codes, so they can become viable in real-world scenarios where low-delay communication is required.

We have also incorporated the subspace error localization in this context; it reasonably improves the error localization, compared to error localization in the PGZ algorithm, and leads to a better MSE. Finally, based on feedback from the decoder, we have introduced rate adaptation in the new framework to compensate for unstable correlation channel and increase the error correction capability of the code if required. Due to rate adaptation, the MSE can decrease significantly for very small compression loss. A more accurate algorithm for error localization is a key to further improve the reconstruction error, as it can be seen from the promising performance of the genie-aided (perfect) error localization in Fig. 8. Further work includes investigation of better error detection and localization algorithms, possibly iterative approach. We will also look at the decoding performance of our system using lattice vector quantization instead of the scalar quantization.

ACKNOWLEDGEMENT

The authors would like to thank the anonymous reviewers of the original manuscript for their thoughtful comments and suggestions that helped improve this paper substantially.

REFERENCES

[1] D. Slepian and J. K. Wolf, "Noiseless coding of correlated information sources," *IEEE Trans. Inf. Theory*, vol. IT-19, pp. 471–480, July 1973.
[2] 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, pp. 1–10, Jan. 1976.
[3] B. Girod, A. M. Aaron, S. Rane, and D. Rebollo-Monedero, "Distributed video coding," *Proc. IEEE*, vol. 93, pp. 71–83, Jan. 2005.
[4] Z. Xiong, A. D. Liveris, and S. Cheng, "Distributed source coding for sensor networks," *IEEE Signal Process. Mag.*, vol. 21, pp. 80–94, Sep. 2004.
[5] S. S. Pradhan and K. Ramchandran, "Distributed source coding using syndromes (DISCUS): Design and construction," *IEEE Trans. Inf. Theory*, vol. 49, pp. 626–643, Mar. 2003.
[6] P. L. Dragotti and M. Gastpar, *Distributed Source Coding: Theory, Algorithms, and Applications*. Academic Press, 2009.
[7] A. D. Liveris, Z. Xiong, and C. N. Georghiades, "Compression of binary sources with side information at the decoder using LDPC codes," *IEEE Commun. Lett.*, vol. 6, pp. 440–442, Oct. 2002.
[8] J. Bajcsy and P. Mitran, "Coding for the Slepian-Wolf problem with turbo codes," in *Proc. GLOBECOM*, vol. 2, pp. 1400–1404, 2001.
[9] A. Aaron and B. Girod, "Compression with side information using turbo codes," in *Proc. DCC*, pp. 252–261, 2002.
[10] E. Akyol, K. Rose, and T. Ramstad, "Optimal mappings for joint source channel coding," in *Proc. ITW*, pp. 1–5, 2010.
[11] E. Akyol and K. Rose, "Optimized analog mappings for distributed source-channel coding," in *Proc. DCC, 2010*, pp. 159–168, 2010.
[12] E. Akyol, K. Viswanatha, K. Rose, and T. Ramstad, "On zero delay source-channel coding," *arXiv:1302.3660*, 2013.
[13] X. Chen and E. Tuncel, "Zero-delay joint source-channel coding for the Gaussian Wyner-Ziv problem," in *ISIT*, pp. 2929–2933, 2011.
[14] T. Marshall Jr., "Coding of real-number sequences for error correction: A digital signal processing problem," *IEEE IEEE J. Sel. Areas Commun.*, vol. 2, pp. 381–392, Mar. 1984.
[15] J. K. Wolf, "Redundancy, the discrete Fourier transform, and impulse noise cancellation," *IEEE Trans. Commun.*, vol. 31, pp. 458–461, Mar. 1983.
[16] V. S. S. Nair and J. A. Abraham, "Real-number codes for fault-tolerant matrix operations on processor arrays," *IEEE Trans. Comput.*, vol. 39, pp. 426–435, Apr. 1990.
[17] F. Marvasti, M. Hasan, M. Echhart, and S. Talebi, "Efficient algorithms for burst error recovery using FFT and other transform kernels," *IEEE Trans. Signal Process.*, vol. 47, pp. 1065–1075, Apr. 1999.
[18] Z. Chen and J. Dongarra, "Numerically stable real number codes based on random matrices," *Computational Science–ICCS*, pp. 115–122, 2005.
[19] R. E. Blahut, *Algebraic Codes for Data Transmission*. New York: Cambridge University Press, 2003.
[20] G. Rath and C. Guillemot, "Subspace algorithms for error localization with quantized DFT codes," *IEEE Trans. Communications*, vol. 52, pp. 2115–2124, Dec. 2004.
[21] 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, pp. 1568–1583, June 2007.
[22] G. Takos and C. N. Hadjicostis, "Determination of the number of errors in DFT codes subject to low-level quantization noise," *IEEE Trans. Signal Process.*, vol. 56, pp. 1043–1054, Mar. 2008.
[23] M. Vaezi and F. Labeau, "Least squares solution for error correction on the real field using quantized DFT codes," in *Proc. EUSIPCO*, pp. 2561–2565, 2012.
[24] V. K. Goyal, J. Kovačević, and J. A. Kelner, "Quantized frame expansions with erasures," *Appl. Comput. Harmonic Anal.*, vol. 10, no. 3, pp. 203–233, 2001.
[25] G. Rath and C. Guillemot, "Frame-theoretic analysis of DFT codes with erasures," *IEEE Trans. Signal Process.*, vol. 52, pp. 447–460, Feb. 2004.
[26] M. Vaezi and F. Labeau, "Extended subspace error localization for rate-adaptive distributed source coding," in *Proc. ISIT*, pp. 2174–2178, 2013.
[27] X. Artigas, J. Ascenso, M. Dalai, S. Klomp, D. Kubasov, and M. Ouaret, "The DISCOVER codec: architecture, techniques and evaluation," in *Proc. PCS*, 2007.
[28] R. Puri and K. Ramchandran, "PRISM: A new robust video coding architecture based on distributed compression principles," in *Proc. Allerton*, 2002.
[29] N. M. Cheung, H. Wang, and A. Ortega, "Sampling-based correlation estimation for distributed source coding under rate and complexity constraints," *IEEE Trans. Image Processing*, vol. 17, no. 11, pp. 2122–2137, 2008.

[30] N. Wernersson and M. Skoglund, "Nonlinear coding and estimation for correlated data in wireless sensor networks," *IEEE Trans. Commun.*, vol. 57, pp. 2932–2939, Oct. 2009.

[31] X. Chen and E. Tuncel, "Low-delay prediction-and transform-based Wyner–Ziv coding," *IEEE Trans. Signal Process.*, vol. 59, pp. 653–666, Feb. 2011.

[32] R. Zamir, S. Shamai, and U. Erez, "Nested linear/lattice codes for structured multiterminal binning," *IEEE Trans. Inf. Theory*, vol. 48, pp. 1250–1276, June 2002.

[33] R. Zamir. The rate loss in the Wyner-Ziv problem. *IEEE Transactions on Information Theory*, 42(6):2073–2084, 1996.

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

[35] Y. Gao and M. Reza Soleymani, "Rate-distortion function for Gauss-Markov source with side information at the decoder," in *25th Biennial Symp. Commun.*, pp. 44–48, 2010.

[36] M. Vaezi and F. Labeau, "Systematic DFT frames: Principle, eigenvalues structure, and applications," *IEEE Trans. Signal Process.*, vol. 61, pp. 3774–3885, Aug. 2013.

[37] M. Vaezi, A. Combernoux, and F. Labeau. "Low-delay joint source-channel coding with side information at the decoder," in *Proc. IEEE Digital Signal Process. Workshop (DSP/SPE)*, pp. 228–232, 2013.

[38] R. O. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE Trans. Antennas Propag.*, vol. 34, pp. 276–280, Mar. 1986.

[39] J. Kovačević and A. Chebira, *An introduction to frames*. Now Publishers, 2008.

[40] F. Bassi, M. Kieffer, and C. Weidmann, "Source coding with intermittent and degraded side information at the decoder," in *Proc. ICASSP*, pp. 2941–2944, 2008.

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

[42] J. Kovačević and A. Chebira, "Life beyond bases: The advent of frames (Part I)," *IEEE Signal Process. Mag.*, vol. 24, pp. 86–104, July 2007.

[43] P. Casazza and J. Kovačević, "Equal-norm tight frames with erasures," *Adv. Comput. Math.*, vol. 18, pp. 387–430, Feb. 2003.

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

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

[46] Q. Xu, V. Stankovic, and Z. Xiong, "Distributed joint source-channel coding of video using raptor codes," *IEEE J. Sel. Areas Commun.*, vol. 25, pp. 851–861, May 2007.

[47] A. D. Liveris, Z. Xiong, and C. N. Georghiades, "Joint source-channel coding of binary sources with side information at the decoder using IRA codes," in *Proc. IEEE Workshop on Multimedia Signal Processing*, pp. 53–56, 2002.

[48] J. Garcia-Frias and Z. Xiong, "Distributed source and joint source-channel coding: From theory to practice," in *Proc. ICASSP*, pp. 1093–1096, 2005.

[49] D. Varodayan, A. Aaron, and B. Girod, "Rate-adaptive codes for distributed source coding," *Signal Process.*, vol. 86, pp. 3123–3130, Nov. 2006.

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

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.