Extended Subspace Error Localization for Rate-Adaptive Distributed Source Coding

Mojtaba Vaezi and Fabrice Labeau

McGill University

International Symposium on Information Theory
Istanbul, Turkey

July 11, 2013

Outline

Real BCH-DFT Codes
Encoding

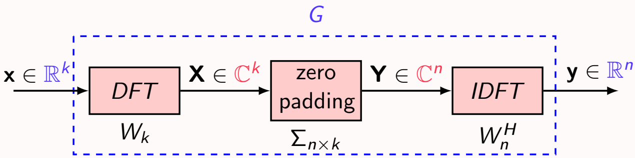


Figure: An (n, k) real BCH-DFT encoding scheme

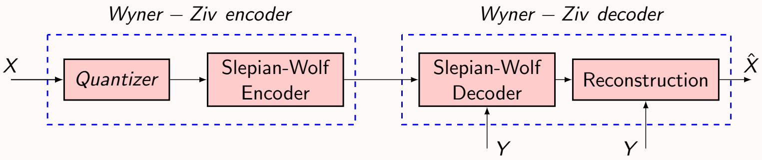
- G consists of k columns from the IDFT matrix of order n
- The remaining $n - k$ columns of the IDFT matrix form H
- $\Sigma_{n\times k}$ inserts $d = n - k$ consecutive zeros in the transform domain $\Longrightarrow$ BCH code
- $\bar{H}$ (the extended parity check matrix) is defined such that $[H^T \mid \bar{H}^T] = W_n$

Lossy DSC

Practical code construction: Binary codes

Problem: Distributed Source Coding (DSC) of continuous-valued sources

- Common approach

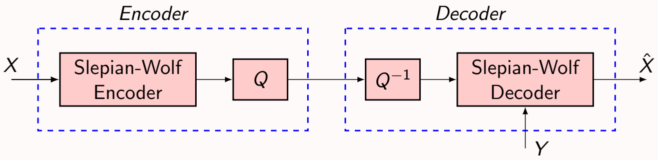


- Binary codes (e.g., LDPC or Turbo codes) for Slepian-Wolf coding
- There are *quantization loss and binning loss*

Lossy DSC

Practical code construction: Real field codes

- Alternative approach



- Similarities and differences
  - There are still coding and quantization losses
  - Coding is before quantization $\Rightarrow$ error correction in the real field (soft redundancy)
- Advantages
  1. Correlation channel model is more realistic
  2. Quantization error can be reduced by a factor of code rate (it vanishes if X and Y are completely correlated)
  3. Better performance w.r.t. delay and complexity

Lossy DSC

Practical code construction

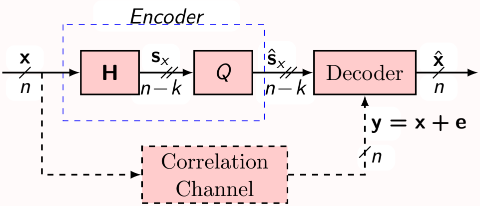


Figure: Wyner-Ziv coding using DFT codes: Syndrome approach.

- Suitable for delay-sensitive networks
- Highly vulnerable to variations in the correlation channel

Lossy DSC

Practical code construction

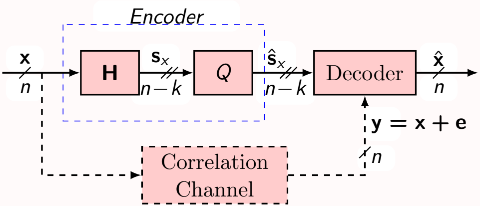


Figure: Wyner-Ziv coding using DFT codes: Syndrome approach.

- Suitable for delay-sensitive networks
- Highly vulnerable to variations in the correlation channel

Motivation for this works

-To make rate-adaptive DSC based on DFT codes

-To improve the decoding algorithm

Outline

1. Error Correction in DFT Codes
   - Coding-Theoretic Approach
   - Subspace-Based Approach
2. Extended Subspace Approach
3. Rate-Adaptive DSC
4. Simulation Results

Outline

1. Error Correction in DFT Codes
   - Coding-Theoretic Approach
   - Subspace-Based Approach
2. Extended Subspace Approach
3. Rate-Adaptive DSC
4. Simulation Results

Some related works

-[Rath and Guillemot, 2004]

-[Duartea and Baraniuk, 2013]

Error Correction in DFT Codes

- Decoding algorithms for a BCH-DFT code:
  1. Detection (to determine the number of errors; $\nu \leq t = \lfloor \frac{n-k}{2} \rfloor$)
  2. Localization (to find the location of errors; $i_1, \ldots, i_\nu$)
  3. Estimation (to calculate the magnitude of errors; $e_{i_1}, \ldots, e_{i_\nu}$)

Error Correction in DFT Codes

- Decoding algorithms for a BCH-DFT code:
  1. Detection (to determine the number of errors; $\nu \leq t = \lfloor \frac{n-k}{2} \rfloor$)
  2. Localization (to find the location of errors; $i_1, \ldots, i_\nu$)
  3. Estimation (to calculate the magnitude of errors; $e_{i_1}, \ldots, e_{i_\nu}$)

$$\mathbf{s} = H\mathbf{r} = H(\mathbf{c} + \mathbf{e}) = H\mathbf{e},$$

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

and $X_p = e^{\frac{j2\pi}{n} i_p}$, $p = 1, \ldots, \nu$.

$$\mathbf{S}_t = \begin{bmatrix} s_1 & s_2 & \ldots & s_t \\ s_2 & s_3 & \ldots & s_{t+1} \\ \vdots & \vdots & \ddots & \vdots \\ s_t & s_{t+1} & \ldots & s_{2t-1} \end{bmatrix}$$

Error Correction in DFT Codes

Coding-Theoretic Approach

1. Detection $(\nu = ?)$
   - $\nu = \mu$ iff $\mathbf{S}_\mu$ is nonsingular but $\mathbf{S}_{\mu+1}$ is singular
2. Localization $(X_i = ?)$
   - Define error-locator polynomial as

$$\Lambda(x) = \prod_{i=1}^{\nu}(1 - xX_i) = \Lambda_0 + \Lambda_1 x + \ldots + \Lambda_\nu x^\nu$$

   - Find $\Lambda_1, \ldots, \Lambda_\nu$ by solving $S_\nu \left[\Lambda_\nu \ldots \Lambda_1\right]^T = -\left[s_{\nu+1} \ldots s_{2\nu}\right]^T$
   - Evaluate $\Lambda(\omega^{-i}), i = 1 \ldots N$, $\omega = e^{j\frac{2\pi}{N}}$, to find the roots
3. Estimation $(Y_i = ?)$
   - Determine error magnitudes by solving the set of following linear equations

$$\begin{bmatrix} s_1 \\ s_2 \\ \vdots \\ s_{2t} \end{bmatrix} = \begin{bmatrix} X_1 & \ldots & X_\nu \\ X_1^2 & \ldots & X_\nu^2 \\ \vdots & \ddots & \vdots \\ X_1^{2t} & \ldots & X_\nu^{2t} \end{bmatrix} \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_{2\nu} \end{bmatrix}$$

Error Correction in DFT Codes

Subspace-Based Approach

- Form the *error-locator matrix* of order m as

$$V_m = \begin{bmatrix} 1 & 1 & \dots & 1 \\ X_1 & X_2 & \dots & X_\nu \\ \vdots & \vdots & \ddots & \vdots \\ X_1^{m-1} & X_2^{m-1} & \dots & X_\nu^{m-1} \end{bmatrix}.$$

- Define the syndrome matrix (for $m = \lceil d/2 \rceil$) by

$$\begin{aligned} S_m &= V_m D V_{d-m+1}^T \\ &= \begin{bmatrix} s_1 & s_2 & \dots & s_{d-m+1} \\ s_2 & s_3 & \dots & s_{d-m+2} \\ \vdots & \vdots & \ddots & \vdots \\ s_m & s_{m+1} & \dots & s_d \end{bmatrix}. \end{aligned}$$

where D is a diagonal matrix of size ν with nonzero diagonal elements $d_p = \frac{1}{\sqrt{n}} e_{i_p} X_p^\alpha, p = 1, \dots, \nu$.

Error Correction in DFT Codes

Subspace-Based Approach

- Eigen-decompose the covariance matrix $R_m = S_m S_m^H$

$$R_m = [U_e \; U_n] \begin{bmatrix} \Delta_e & \mathbf{0} \\ \mathbf{0} & \Delta_n \end{bmatrix} [U_e \; U_n]^H,$$

-Δ_e and Δ_n contain the ν largest and $m-\nu$ smallest eigenvalues
-U_e and U_n contain the eigenvectors corresponding to Δ_e and Δ_n

- The columns in U_e span the *channel-error subspace* spanned by V_m thus, $U_e^H U_n = \mathbf{0} \Rightarrow V_m^H U_n = \mathbf{0}$
- Let $\mathbf{v} = [1, x, x^2, \ldots, x^{m-1}]^T$ where x is a complex variable, then

$$F(x) \triangleq \sum_{j=1}^{m-\nu} \mathbf{v}^H \mathbf{u}_{n,j} = \sum_{j=1}^{m-\nu} \sum_{i=0}^{m-1} f_{ji} x^i.$$

$F(x)$ is sum of $m-\nu$ polynomials $\{f_{ji}\}_{j=1}^{m-\nu}$ of order $m-1$.

Error Correction in DFT Codes

Subspace-Based Approach

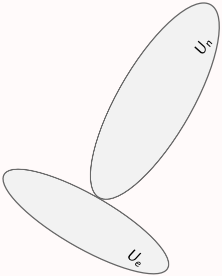


Figure: Subspace method: graphical representation

Error Correction in DFT Codes

Subspace-Based Approach

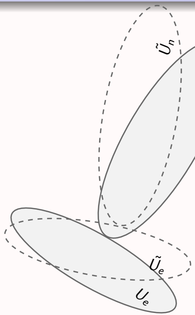


Figure: Subspace method: graphical representation

Error Correction in DFT Codes

Subspace-Based Approach

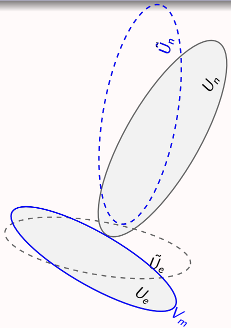


Figure: Subspace method: graphical representation

Error Correction in DFT Codes

Subspace-Based Approach

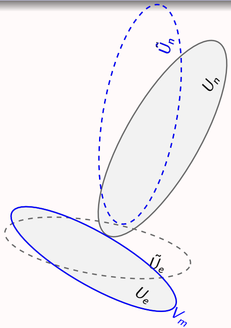


Figure: Subspace method: graphical representation

Subspace vs. coding-theoretic method

There are $m - \nu = \lceil \frac{d}{2} \rceil - \nu$ polynomials rather than just one, and they have higher degrees of freedom

$\Rightarrow$ Subspace method performs better than the coding-theoretic approach

Extended Subspace Decoding

Motivation

Main idea:

Increasing the dimension of the estimated noise subspace $\Rightarrow$ the number of polynomials with linearly independent coefficients and/or their degree grow.

Extended Subspace Decoding

Motivation

Main idea:

Increasing the dimension of the estimated noise subspace $\Rightarrow$ the number of polynomials with linearly independent coefficients and/or their degree grow.

Extended Subspace Decoding
Motivation

Main idea:

Increasing the dimension of the estimated noise subspace ⇒ the number of polynomials with linearly independent coefficients and/or their degree grow.

Construction:

The extended syndrome matrix S'_m is defined for $d' > d$, and similar to S_m it is decomposable as

$$S'_m = V_m D V^T_{d'-m+1}.$$

To form S'_m we need d' syndrome samples while we only have d samples.

$$s'_m = \frac{1}{\sqrt{n}} \sum_{p=1}^{\nu} e_{i_p} X_p^{\alpha-1+m}, \quad m = 1, \ldots, d',$$

Extended Subspace Decoding

Extended Syndrome

$$s'_m = \begin{cases} s_m, & 1 \leq m \leq d, \\ \bar{s}_{m-d}, & d < m \leq d', \end{cases}$$

where $\bar{\mathbf{s}} = \bar{H}\mathbf{e}$, is the extended syndrome of error.
Recal: $\bar{H}$ consists of those k columns of the IDFT matrix of order n not used in H (used in G).

Extended Subspace Decoding

Extended Syndrome

$$s'_m = \begin{cases} s_m, & 1 \leq m \leq d, \\ \bar{s}_{m-d}, & d < m \leq d', \end{cases}$$

where $\bar{\mathbf{s}} = \bar{H}\mathbf{e}$, is the extended syndrome of error.
Recal: $\bar{H}$ consists of those k columns of the IDFT matrix of order n not used in H (used in G).

Q:

How can we compute $\bar{\mathbf{s}}$?

Extended Subspace Decoding

Extended Syndrome

$$s'_m = \begin{cases} s_m, & 1 \le m \le d, \\ \bar{s}_{m-d}, & d < m \le d', \end{cases}$$

where $\bar{\mathbf{s}} = \bar{H}\mathbf{e}$, is the extended syndrome of error.

Recal: $\bar{H}$ consists of those k columns of the IDFT matrix of order n not used in H (used in G).

Q:

How can we compute $\bar{\mathbf{s}}$?

Extended Subspace Decoding

Extended Syndrome

$$s'_m = \begin{cases} s_m, & 1 \leq m \leq d, \\ \bar{s}_{m-d}, & d < m \leq d', \end{cases}$$

where $\bar{\mathbf{s}} = \bar{H}\mathbf{e}$, is the extended syndrome of error.

Recal: $\bar{H}$ consists of those k columns of the IDFT matrix of order n not used in H (used in G).

Q:

How can we compute $\bar{\mathbf{s}}$?

Let us try

$$\bar{H}\mathbf{r} = \bar{H}\mathbf{c} + \bar{H}\mathbf{e} \neq \bar{H}\mathbf{e}$$

So to have $\bar{H}\mathbf{r} = \bar{\mathbf{s}}$, either $\bar{H}\mathbf{c}$ must vanish or we should remove it.

Extended Subspace Decoding

Extended Syndrome

- $\bar{H}\mathbf{c} = \mathbf{0}$ could happen in the special case of rate $\frac{1}{2}$ codes when all error indices are even
- In general, we need to find a way to remove $\bar{H}\mathbf{c}$

We exploit the gain from the extended subspace decoding by transmitting extra samples $\Rightarrow$ Rate-adaptive DFT codes

Extended Subspace Decoding

Extended Syndrome

- $\bar{H}\mathbf{c} = \mathbf{0}$ could happen in the special case of rate $\frac{1}{2}$ codes when all error indices are even
- In general, we need to find a way to remove $\bar{H}\mathbf{c}$

We exploit the gain from the extended subspace decoding by transmitting extra samples $\Rightarrow$ Rate-adaptive DFT codes

Extended Subspace Decoding

Extended Syndrome

- $\bar{H}\mathbf{c} = \mathbf{0}$ could happen in the special case of rate $\frac{1}{2}$ codes when all error indices are even
- In general, we need to find a way to remove $\bar{H}\mathbf{c}$

We exploit the gain from the extended subspace decoding by transmitting extra samples $\Rightarrow$ Rate-adaptive DFT codes

Applications

1. Rate-adaptive DSC (syndrome & parity approaches)
2. Rate-adaptive channel coding
3. Rate-adaptive distributed joint source-channel coding

Rate-Adaptive DSC

Syndrome Approach

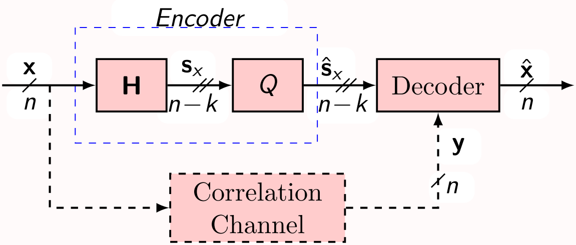


Figure: The Wyner-Ziv coding using DFT codes: Syndrome approach.

Rate-Adaptive DSC

Syndrome Approach

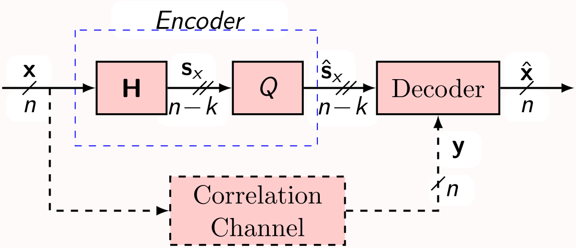


Figure: The Wyner-Ziv coding using DFT codes: Syndrome approach.

Rate Adaptation:

1. **Decoder:** Request for extra syndrome samples
2. **Encoder:** Transmit $\bar{\mathbf{s}}_x = \bar{H}\mathbf{x}$ sample by sample
3. **Decoder:** Compute $\bar{\mathbf{s}}_y = \bar{H}\mathbf{y} = \bar{\mathbf{s}}_x + \bar{\mathbf{s}}_e$ and $\bar{\mathbf{s}}_e = \bar{\mathbf{s}}_y - \bar{\mathbf{s}}_x$
4. **Decoder:** Use the extended subspace decoding

Rate-Adaptive DSC

Parity Approach

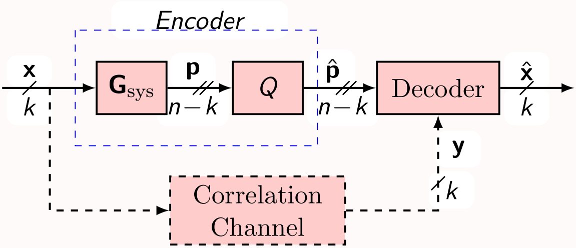


Figure: The Wyner-Ziv coding using DFT codes: Parity approach.

Rate-Adaptive DSC

Parity Approach

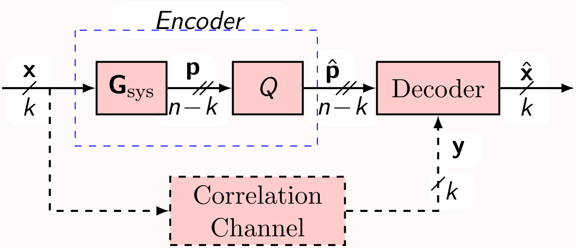


Figure: The Wyner-Ziv coding using DFT codes: Parity approach.

Rate Adaptation:

1. **Parity Puncturing:** Performance is poor
2. **Syndrome Augmentation:** Very similar to the syndrome-based DSC

Numerical results

Syndrome-based DSC

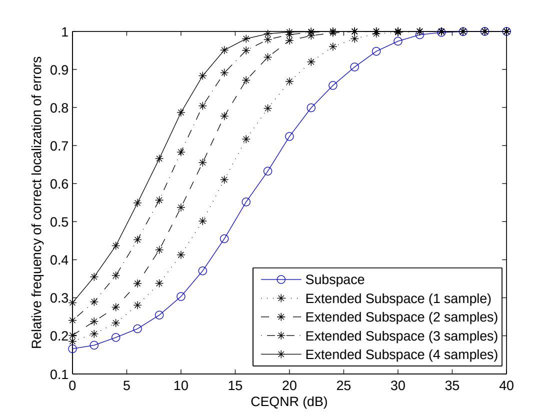


Figure: Rate-adaptation using a (10,5) DFT code and extended subspace method. The the code rate is increased from 0.5 to 0.9 by a step of 0.1.

Numerical results

Syndrome-based DSC

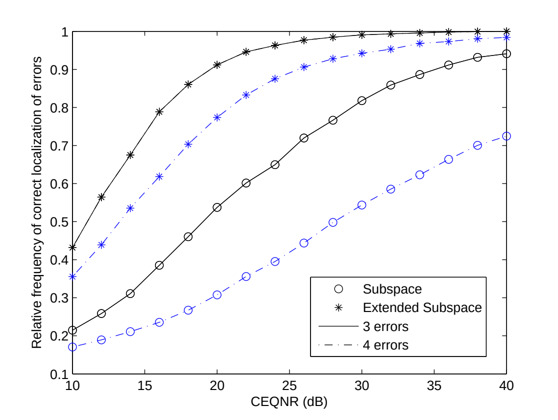


Figure: Rate-adaptation for a (17,9) DFT code based on 4 additional syndrome samples.

Conclusions

Summary: The extended subspace algorithm of DFT codes

- Improves decoding at the expense of increasing the code-rate
- Is suitable for rate-adaptive DSC
- Makes possible to correct more than $t = \lfloor \frac{n-k}{2} \rfloor$ errors

The gain comes from diminishing the effect of quantization error by averaging several error localization polynomials.

Generalization:

- To further improve the subspace decoding without increasing the code rate.

Thank you!