Systematic Code Design for Combined Channel Estimation and Error Correction and Its Maximum-Likelihood Decoding

Chia-Lung Wu

Advisor: Prof. Po-Ning Chen

Outline

- Introduction
- Technical Background
- Code Designs for Frequency-Selective Block Fading Channels
- Codes Designs for Frequency-Selective Varying Fading Channels
- Code Designs for Frequency-Nonselective Varying Fading Channels
- A Systematic Space-Time Code Design
- Conclusion and future work

Introduction

Introduction

- A typical receiver in a wireless communication system performs **channel estimation** and **data estimation** separately.
- In 2002, Skoglund *et al.* provided a milestone evidence that a code design that jointly considers channel estimation and error correction is able to counter multi-path block fading more efficiently than the approach with a separate error-correcting code and channel estimation scheme.
  - Through simulations, they found that a communication system using these nonlinear codes can outperform a typical communication system with perfect channel estimation by 2 dB.
- One of the drawbacks of these joint estimation codes found by computer search is that they lack a systematic structure, and can therefore be decoded only by an operation-intensive exhaustive search.

Contributions

- In this dissertation, this drawback was resolved firstly by discovering that regardless of the fading statistics, the codeword that maximizes the system signal-to-noise ratio (SNR) must be orthogonal to the delayed version of itself. We termed this property *self-orthogonality*.
- We secondly found that the code that consists of properly chosen self-orthogonal codewords has a performance comparable to that of the simulated-annealing-based computer-searched code.
- We then derived a recursive maximum-likelihood metric for self-orthogonal codewords for use of the priority-first search algorithm, resulting in a decoding complexity significantly smaller than that required by exhaustive decoding.
- Codes for various channels such as frequency-selective block fading and frequency-selective and nonselective varying fading, as well as MIMO fading channels, are provided.

Technical Background

Generalized Likelihood Ratio Test (GLRT)

The single-input-single-output (SISO) time-discrete system can be modelled as

$$\boldsymbol{y} = \mathbb{B}\boldsymbol{h} + \boldsymbol{n}$$

where

$$\mathbb{B} \triangleq \begin{bmatrix} b_1 & 0 & \cdots & 0 \\ \vdots & b_1 & \ddots & \vdots \\ b_N & \vdots & \ddots & 0 \\ 0 & b_N & \ddots & b_1 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & b_N \end{bmatrix}_{L \times P},$$

is a $L \times P$ signaling matrix that emulates the convolution operation with channel coefficient $\boldsymbol{h}$, and every non-zero entity of $\mathbb{B}$ is a bipolar symbol. Here, $\boldsymbol{h}$ is the $P \times 1$ channel coefficient vector and $\boldsymbol{n}$ is a $L \times 1$ noise vector, where $L = N + P - 1$.

Generalized Likelihood Ratio Test (GLRT)

- Under the condition that transmitted signal $\mathbb{B}$ is equally likely, the maximum-likelihood (ML) detector is

$$\hat{\mathbb{B}} = \arg\max_{\mathbb{B}} \Pr(\boldsymbol{y}|\mathbb{B})$$

  Obviously, the ML decision requires the knowledge of probability of $\boldsymbol{y}$ given $\mathbb{B}$.

- For the case that $\boldsymbol{n}$ is zero-mean Gaussian distributed with covariance matrix $\sigma^2\mathbb{I}_L$, and that the receiver has perfect knowledge about $\boldsymbol{h}$, we know that

$$\Pr(\boldsymbol{y}|\mathbb{B}, \boldsymbol{h}) = \frac{1}{|\pi\sigma^2\mathbb{I}_L|} e^{-\|\boldsymbol{y}-\mathbb{B}\boldsymbol{h}\|^2/\sigma^2}$$

Generalized Likelihood Ratio Test (GLRT)

- Since $\boldsymbol{h}$ is completely unknown to the receiver (and also the transmitter), the decision maker at the receiver end can only rely on the well-known generalized likelihood ratio test (GLRT):

$$\begin{aligned}
\hat{\mathbb{B}} &= \arg\max_{\mathbb{B}} \max_{\boldsymbol{h}} \Pr(\boldsymbol{y}|\mathbb{B}, \boldsymbol{h}) \\
&= \arg\max_{\mathbb{B}} \Pr(\boldsymbol{y}|\mathbb{B}, \hat{\boldsymbol{h}}) \\
&= \arg\max_{\mathbb{B}} \frac{1}{|\pi\sigma^2\mathbb{I}|} e^{-\|\boldsymbol{y}-\mathbb{B}\hat{\boldsymbol{h}}\|^2/\sigma^2} \\
&= \arg\min_{\mathbb{B}} \|\boldsymbol{y} - \mathbb{B}(\mathbb{B}^H\mathbb{B})^{-1}\mathbb{B}^H\boldsymbol{y}\|^2 \\
&= \arg\min_{\mathbb{B}} \|\boldsymbol{y} - \mathbb{P}_B\boldsymbol{y}\|^2
\end{aligned}$$

where the least square estimate of $\boldsymbol{h}$ for a given code word $\mathbb{B}$ is

$$\hat{\boldsymbol{h}} = (\mathbb{B}^H\mathbb{B})^{-1}\mathbb{B}^H\boldsymbol{y},$$

and $\mathbb{P}_B \triangleq \mathbb{B}(\mathbb{B}^H\mathbb{B})^{-1}\mathbb{B}^H$.

Generalized Likelihood Ratio Test (GLRT)

- GLRT is actually the best decision under completely blind in $\boldsymbol{h}$; so some researchers describe it an **optimal** decision in the above setting for which some researchers do not agree with this naming.
  - The approximate PEP of the ML detection with known $\boldsymbol{h}$ is

$$\Pr\left(\hat{\mathbb{B}} = \mathbb{B}(j) \middle| \mathbb{B}(i) \text{ transmitted}\right) \approx \frac{K_{i,j}}{|\boldsymbol{C}_h|\,|\mathbb{B}(i)^H(\mathbb{I} - \mathbb{P}_{B(j)})\mathbb{B}(i)|}$$

where

$$K_{i,j} = \begin{cases} \dfrac{\det|\mathbb{B}(i)^H\mathbb{B}(i)|}{\det|\mathbb{B}(j)^H\mathbb{B}(j)|}\displaystyle\sum_{k=0}^{P-1}\binom{2P-1-k}{P-1}\frac{1}{k!}\left(\ln\frac{\det|\mathbb{B}(j)^H\mathbb{B}(j)|}{\det|\mathbb{B}(i)^H\mathbb{B}(i)|}\right)^k, \\ \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \text{if } \dfrac{\det|\mathbb{B}(i)^H\mathbb{B}(i)|}{\det|\mathbb{B}(j)^H\mathbb{B}(j)|} \le 1 \\ \displaystyle\sum_{k=0}^{P}\binom{2P-1-k}{P-1}\frac{1}{k!}\left(\ln\frac{\det|\mathbb{B}(i)^H\mathbb{B}(i)|}{\det|\mathbb{B}(j)^H\mathbb{B}(j)|}\right)^k, \\ \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \text{otherwise.} \end{cases}$$

Here, assume $0^0 = 1$ for convenience.

Generalized Likelihood Ratio Test (GLRT)

- For GLRT detection with unknown $\boldsymbol{h}$, its approximate PEP is

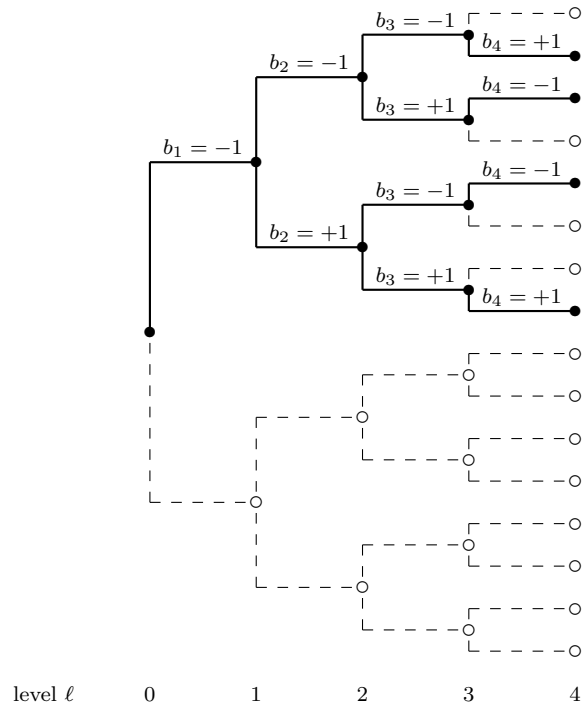
$$\Pr\left(\hat{\mathbb{B}} = \mathbb{B}(j) \middle| \mathbb{B}(i) \text{ transmitted}\right) \approx \frac{\binom{2P-1}{P}}{|\boldsymbol{C}_h|\,|\mathbb{B}(i)^H(\mathbb{I} - \mathbb{P}_{B(j)})\mathbb{B}(i)|}$$

- It can be seen that when $|\mathbb{B}(i)^H\mathbb{B}(i)| = |\mathbb{B}(j)^H\mathbb{B}(j)|$, the approximate PEP of ML-with-known-$\boldsymbol{h}$ and GLRT detections are the same.
- Accordingly, the performances of the GLRT detection in our coding system with

$$|\mathbb{B}(i)^H\mathbb{B}(i)| = |\mathbb{B}(j)^H\mathbb{B}(j)|$$

  is equivalent to the ML detection with known $\boldsymbol{h}$.

The ML Priority-First Search Decoding Algorithm



$$\mathbb{B}^T\mathbb{B} = \mathbb{G}$$

$$\boldsymbol{b} \equiv \mathbb{B} \equiv \mathbb{P}_B$$

Step 1. **(Initialization)** *Load the stack with the path that ends at the original node.*

Step 2. **(Evaluation)** *Evaluate the metric values of the successor paths of the current top path in the stack. Then delete this top path from the stack.*

Step 3. **(Sorting)** *Insert the successor paths obtained in Step 2 into the stack such that the paths in the stack are ordered according to their ascending metric values.*

Step 4. **(Loop)** *If the top path in the stack ends at a terminal node in the code tree, output the labels corresponding to the top path, and the algorithm stops; otherwise, go to Step 2.*

The ML Priority-First Search Decoding Algorithm

A sufficient condition under which the priority-first search algorithm is guaranteed to locate the path with the smallest metric among all paths.

Lemma If the metric f is nondecreasing along every path $\boldsymbol{b}_{(\ell)}$ in the code tree, i.e.,

$$f\left(\boldsymbol{b}_{(\ell)}\right) \leq \min_{\left\{\tilde{\boldsymbol{b}} \in \mathcal{C} \,:\, \tilde{\boldsymbol{b}}_{(\ell)} = \boldsymbol{b}_{(\ell)}\right\}} f(\tilde{\boldsymbol{b}}), \tag{1}$$

then the priority-first search algorithm always yields the code path with the smallest metric value among all code paths of $\mathcal{C}$.

The ML Priority-First Search Decoding Algorithm

- It is convenient to represent metric f as the sum of two components:

$$f(\boldsymbol{b}_{(\ell)}) \triangleq g(\boldsymbol{b}_{(\ell)}) + \varphi(\boldsymbol{b}_{(\ell)}).$$

  - The first component g is directly defined based on the maximum-likelihood metric such that

$$\arg\min_{\boldsymbol{b}\in\mathcal{C}} g(\boldsymbol{b}) = \arg\min_{\boldsymbol{b}\in\mathcal{C}} \|\boldsymbol{y} - \mathbb{P}_B\boldsymbol{y}\|^2.$$

  - The second component φ is designed to validate (1) with $\varphi(\boldsymbol{b}) = 0$ for any $\boldsymbol{b} \in \mathcal{C}$.
  - Then from $f(\boldsymbol{b}) = g(\boldsymbol{b}) + \varphi(\boldsymbol{b}) = g(\boldsymbol{b})$ for all $\boldsymbol{b} \in \mathcal{C}$, the desired maximum-likelihood priority-first search decoding algorithm is established.

The ML Priority-First Search Decoding Algorithm

- A typical interpretation of the so-called *heuristic function* φ is that it helps predict a future route from the end node of the current path to a terminal node.
- Notably, the design of the heuristic function φ that validates condition (1) is not unique. Different designs may result in variations in computational complexity.

Code Designs for Frequency-Selective Block Fading Channels

Review and Contributions

- No systematic code constructions have been proposed for joint channel and data estimation in quasi-static fading channels.
  - Efforts have mostly been invested in computer searches for codes that counter channel fading.
  - The decoding of such structureless computer-searched codes thus becomes an engineering challenge.
- In 2003, Skoglund *et al.* relied on simulated annealing to search for nonlinear binary block codes suitable for joint channel and data estimation in quasi-static fading channels.
  - Although the operating SNR for the code search was set at 10 dB, their codes perform well under a wide range of SNRs.
  - The mismatch in the relative powers of different channel coefficients has no big effect on the resulting performance either.
  - Their results indicate that the nonlinear estimation codes can outperform a typical linear error correcting code operated with a perfect channel estimator.

Review and Contributions

- In 2005, Coskun and Chugg replaced the PEP sum by a properly defined pairwise distance measure between two codewords, and proposed a suboptimal greedy algorithm to speed up the code search process.
- In 2007, Giese and Skoglund re-applied their original idea to single and multiple-antenna systems.

Review and Contributions

Contributions:

- We establish a design constraint and show that this constraint can maximize the system SNR regardless of the fading statistics.
  - As it happens that Skoglund's codes satisfy this constraint, their insensitivity to SNR and channel mismatch now find a theoretical support.
- By the systematic structure of our codes, we can derive a recursive ML decoding metric for the priority-first search decoding.
  - The decoding complexity is significantly decreased in contrast to that of the exhaustive decoder required by the structureless computer-searched codes.

System Model and ML Decoding Criterion

- $\boldsymbol{b} = [b_1, \ldots, b_N]^T$ of an (N, K) code $\mathcal{C}$ is transmitted over a block fading channel of memory order $P-1$, where each $b_j \in \{-1, +1\}$.
- Channel coefficients $\boldsymbol{h} = [h_1, \ldots, h_P]^T$ are *constant* within a coding block of length $L = N + P - 1$.

By letting the codeword matrix be

$$\mathbb{B} \triangleq \begin{bmatrix} b_1 & 0 & \cdots & 0 \\ \vdots & b_1 & \ddots & \vdots \\ b_N & \vdots & \ddots & 0 \\ 0 & b_N & \ddots & b_1 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & b_N \end{bmatrix}_{L \times P},$$

the complex-valued received vector $\boldsymbol{y}$ is given by

$$\boldsymbol{y} = \mathbb{B}\boldsymbol{h} + \boldsymbol{n}, \tag{2}$$

where $\boldsymbol{n} \sim \mathcal{CN}(\mathbf{0}, \sigma_n^2 \mathbb{I}_L)$ and $\mathbb{I}_L$ is the $L \times L$ identity matrix.

System Model and ML Decoding Criterion

Assumption:

- Both transmitter and receiver know nothing about the channel coefficients $\boldsymbol{h}$, but have knowledge of the multi-path parameter P.
- There are adequate guard periods between consecutive encoding blocks such that zero interblock interference is guaranteed.

The LS estimate of the channel coefficients $\boldsymbol{h}$ for a given $\boldsymbol{b}$ equals

$$\hat{\boldsymbol{h}} = (\mathbb{B}^T\mathbb{B})^{-1}\mathbb{B}^T\boldsymbol{y}.$$

The joint ML decision becomes

$$\hat{\boldsymbol{b}} = \arg\min_{\boldsymbol{b}\in\mathcal{C}} \min_{\boldsymbol{h}} \|\boldsymbol{y}-\mathbb{B}\boldsymbol{h}\|^2 = \arg\min_{\boldsymbol{b}\in\mathcal{C}} \|\boldsymbol{y}-\mathbb{B}\hat{\boldsymbol{h}}\|^2 = \arg\min_{\boldsymbol{b}\in\mathcal{C}} \|\boldsymbol{y}-\mathbb{P}_B\boldsymbol{y}\|^2,$$

where $\mathbb{P}_B \triangleq \mathbb{B}(\mathbb{B}^T\mathbb{B})^{-1}\mathbb{B}^T$.[1]

[1] The mapping from a codeword $\boldsymbol{b}$ to the corresponding transformed codeword $\mathbb{P}_B$ is not one-to-one unless b_1 is fixed. We always set $b_1 = -1$ for the codes we construct. Under the setting, it is obvious that the largest code rate attainable by our code design is $(N-1)/N$.

Code Constraint that Maximizes the Average SNR Regardless of Channel Statistics

The average SNR conditional on the input $\mathbb{B}$ satisfies

$$\frac{E[\|\mathbb{B}\boldsymbol{h}\|^2|\mathbb{B}]}{E[\|\boldsymbol{n}\|^2]} = \frac{N}{L\sigma_n^2}\mathrm{tr}\left[E[\boldsymbol{h}\boldsymbol{h}^H]\left(\frac{1}{N}\mathbb{B}^T\mathbb{B}\right)\right]. \tag{3}$$

The average SNR can be as worse as

$$\min_{\{\boldsymbol{h}\,:\,\mathrm{tr}(E[\boldsymbol{h}\boldsymbol{h}^H])=\tau\}} \frac{E[\|\mathbb{B}\boldsymbol{h}\|^2|\mathbb{B}]}{E[\|\boldsymbol{n}\|^2]},$$

where τ is a certain power level on the channel coefficients $\boldsymbol{h}$. It can be upper-bounded by a constant, i.e.,

$$\min_{\{\boldsymbol{h}\,:\,\mathrm{tr}(E[\boldsymbol{h}\boldsymbol{h}^H])=\tau\}} \frac{E[\|\mathbb{B}\boldsymbol{h}\|^2|\mathbb{B}]}{E[\|\boldsymbol{n}\|^2]} \leq \frac{E[\|\mathbb{B}\tilde{\boldsymbol{h}}\|^2|\mathbb{B}]}{E[\|\boldsymbol{n}\|^2]} = \left(\frac{N}{L\sigma_n^2}\right)\tau,$$

where the above inequality holds since an upper bound can be resulted by taking any $\boldsymbol{h}$ that satisfies $\mathrm{tr}(E[\boldsymbol{h}\boldsymbol{h}^H]) = \tau$ into $E[\|\mathbb{B}\boldsymbol{h}\|^2|\mathbb{B}]/E[\|\boldsymbol{n}\|^2]$, and here we take $\tilde{\boldsymbol{h}}$ to be zero-mean i.i.d. with $\mathrm{tr}(E[\tilde{\boldsymbol{h}}\tilde{\boldsymbol{h}}^H]) = \tau$.

Code Construction

- This constant SNR bound can be achieved even if the system is totally blind on channel coefficients $\boldsymbol{h}$ (as well as the power level τ), when the codeword is designed to be *self-orthogonal* in the sense that

$$\frac{1}{N}\mathbb{B}^T\mathbb{B} = \mathbb{I}_P. \tag{4}$$

- Condition (4) actually has an operational meaning. It ensures that every codeword is orthogonal to the shifted version of itself, and hence temporal diversity can be implicitly realized even under completely no knowledge on channel statistics.

- We henceforth say that codewords constrained on (4) maximize the average SNR attainable regardless of the statistics of $\boldsymbol{h}$.

Code Construction

- Unfortunately, a codeword sequence satisfying (4) is only guaranteed to exist for $P = 2$ with N odd (and trivially, for $P = 1$).
- In some other cases, such as $P = 3$, one can only design codes to approximately satisfy (4). For example,

$$\frac{1}{N}\mathbb{B}^T\mathbb{B} = \frac{1}{N}\begin{bmatrix} N & \pm 1 & 0 \\ \pm 1 & N & \pm 1 \\ 0 & \pm 1 & N \end{bmatrix} \quad \text{for } N \text{ even,}$$

and

$$\frac{1}{N}\mathbb{B}^T\mathbb{B} = \frac{1}{N}\begin{bmatrix} N & 0 & \pm 1 \\ 0 & N & 0 \\ \pm 1 & 0 & N \end{bmatrix} \quad \text{for } N \text{ odd.}$$

- We therefore relax (4) and allow some off-diagonal entries in $\mathbb{B}^T\mathbb{B}$ to be either -1 or 1 whenever it is impossible to strictly satisfy (4). We will denote such a matrix as $\mathbb{G}$.
- This particular structure of $\mathbb{G}$ can really be observed in the simulated-annealing-based computer-searched codes.

Code Construction

- For $4 \leq N \leq 18$ and N even, the best computer-searched half-rate codes that minimize the sum of PEPs under complex zero-mean Gaussian distributed $\boldsymbol{h}$ with $E[\boldsymbol{h}\boldsymbol{h}^H] = (1/2)\mathbb{I}_P$ and $P = 2$ all satisfy the relation

$$\mathbb{B}^T\mathbb{B} = \begin{bmatrix} N & \pm 1 \\ \pm 1 & N \end{bmatrix}. \tag{5}$$

- We have also obtained and examined Skoglund's computer-searched code used for $N = 22$, and found as anticipated that every codeword satisfies (5).

- Crozier et al suggest that for an optimal channel estimation, the training sequences $\boldsymbol{b}$ can be chosen such that $\mathbb{B}^T\mathbb{B}$ is proportional to $\mathbb{I}_P$.

- In 2007, Giese *et al.* remark that a code sequence with a certain aperiodic autocorrelation property possibly could be exploited in future code design approaches.

Code Construction

- The joint ML decision can be reformulated as

$$\begin{aligned}
\hat{\boldsymbol{b}} &= \arg\min_{\boldsymbol{b}\in\mathcal{C}} (\boldsymbol{y} - \mathbb{P}_B\boldsymbol{y})^H(\boldsymbol{y} - \mathbb{P}_B\boldsymbol{y}) \\
&= \arg\min_{\boldsymbol{b}\in\mathcal{C}} -\mathrm{tr}(\mathbb{P}_B\boldsymbol{y}\boldsymbol{y}^H) \qquad (6)\\
&= \arg\min_{\boldsymbol{b}\in\mathcal{C}} \big(\|\mathrm{vec}(\boldsymbol{y}\boldsymbol{y}^H)\|^2 - \mathrm{vec}(\mathbb{P}_B)^T\mathrm{vec}(\boldsymbol{y}\boldsymbol{y}^H) \\
&\quad -\mathrm{vec}(\boldsymbol{y}\boldsymbol{y}^H)^H\mathrm{vec}(\mathbb{P}_B) + \|\mathrm{vec}(\mathbb{P}_B)\|^2\big) \\
&= \arg\min_{\boldsymbol{b}\in\mathcal{C}} \|\mathrm{vec}(\boldsymbol{y}\boldsymbol{y}^H) - \mathrm{vec}(\mathbb{P}_B)\|^2.
\end{aligned}$$

- We can then bound the ML error probability by the union bound:

$$\begin{aligned}
P_\mathrm{e} \le \frac{1}{2^K}\sum_{i=1}^{2^K}\sum_{\substack{j=1\\ j\neq i}}^{2^K} \Pr\big(&\|\mathrm{vec}(\boldsymbol{y}\boldsymbol{y}^H) - \mathrm{vec}(\mathbb{P}_{B_j})\|^2 \\
&< \|\mathrm{vec}(\boldsymbol{y}\boldsymbol{y}^H) - \mathrm{vec}(\mathbb{P}_{B_i})\|^2 \big| \, \boldsymbol{b}_i \text{ transmitted}\big),
\end{aligned}$$

where $\boldsymbol{b}_i$ is the ith codeword of an (N, K) block code, and $\mathbb{P}_{B_i}$ denotes the equivalent ith codeword in the $\mathbb{P}_B$-domain.

Code Construction

- Since $\mathbb{P}_B = \mathbb{B}(\mathbb{B}^T\mathbb{B})^{-1}\mathbb{B}^T = \frac{1}{N}\mathbb{B}\mathbb{B}^T$, the PEP-based upper bound then suggests that a good self-orthogonal code design should have an adequately large pairwise Euclidean distance

$$\left\|\mathrm{vec}(\mathbb{B}_i\mathbb{B}_i^T) - \mathrm{vec}(\mathbb{B}_j\mathbb{B}_j^T)\right\|^2 \tag{7}$$

  between all codeword pairs $\mathbb{B}_i$ and $\mathbb{B}_j$, where $\mathbb{B}_i$ is the equivalent ith codeword in the $\mathbb{B}$-domain.

- This is conceptually equivalent to a uniform pick of codewords in a set of self-orthogonal binary sequences.

- The uniform pick principle then suggests that

$$\frac{|\mathcal{C}(b_1, b_2, \ldots, b_{\ell-1}, b_\ell = 1)|}{|\mathcal{A}(b_1, b_2, \ldots, b_{\ell-1}, b_\ell = 1|\mathbb{G})|} \approx \frac{|\mathcal{C}(b_1, b_2, \ldots, b_{\ell-1}, b_\ell = -1)|}{|\mathcal{A}(b_1, b_2, \ldots, b_{\ell-1}, b_\ell = -1|\mathbb{G})|},$$

  where $\mathcal{C}(\boldsymbol{b}_{(\ell)})$ is the set of all codewords whose first ℓ bits equal $b_1, b_2, \ldots, b_\ell$, and $\mathcal{A}(\boldsymbol{b}_{(\ell)}|\mathbb{G})$ is the set of all binary sequences of length N, whose first ℓ bits equal $b_1, b_2, \ldots, b_\ell$, and whose $\mathbb{B}$-representation satisfies $\mathbb{B}^T\mathbb{B} = \mathbb{G}$.

Code Construction

Our goal

- We need an efficient and systematic way to generate the successor paths of the top path.
- We would like to have a code tree that can be spanned in an on-the-fly or bit-by-bit fashion.

Exemplified Code Design Algorithm for Channels of Memory Order One

Here, We provide an exemplified code design algorithm based on the uniform pick principle for channels of memory order 1, namely, $P = 2$.

- For $\theta \in \{-1, 0, +1\}$, we define

$$\mathbb{G}_\theta \triangleq \begin{bmatrix} N & \theta \\ \theta & N \end{bmatrix}.$$

  Note that when $\mathbb{B}^T\mathbb{B} = \mathbb{G}_0$ $(= N \cdot \mathbb{I}_2)$ cannot be satisfied as aforementioned for N even, $\mathbb{G}_{-1}$ and $\mathbb{G}_1$ will be used instead to define the relaxed self-orthogonal codewords.

- In such case, the uniform pick principle again suggests that half of the codewords should be uniformly drawn from binary sequences satisfying $\mathbb{B}^T\mathbb{B} = \mathbb{G}_{-1}$, and the other half of codewords are selected according to $\mathbb{B}^T\mathbb{B} = \mathbb{G}_1$.

Code Construction

Step 1. Input the index i of the requested codeword in the (N, K) block code, where $0 \leq i \leq 2^K - 1$.

Step 2. Set $\Theta = \{0\}$ for N odd, and $\Theta = \{-1, 1\}$ for N even. Also, set $\theta = ((N+1) \bmod 2) \cdot (-1)^{\lceil (i+1)/(2^K/|\Theta|) \rceil}$. Compute

$$\Delta_\theta = \frac{|\mathcal{A}(b_1 = -1|\mathbb{G}_\theta)| - 1}{2^K/|\Theta| - 1} \quad \text{for } \theta \in \Theta. \tag{8}$$

Initialize $b_1 = -1$, $\ell = 1$ and $\rho = \lfloor (i \bmod (2^K/|\Theta|)) \cdot \Delta_\theta \rfloor$. Let the minimum sequence index $\rho_{\min} = 0$.

Step 3. Execute $\ell = \ell + 1$, and compute $\gamma_\ell = |\mathcal{A}(\boldsymbol{b}_{(\ell-1)}, b_\ell = -1|\mathbb{G}_\theta)|$.

If $\rho < \rho_{\min} + \gamma_\ell$, then choose the next code bit $b_\ell = -1$; otherwise choose the next code bit $b_\ell = 1$, and readjust $\rho_{\min} = \rho_{\min} + \gamma_\ell$.

Step 4. If $\ell = N$, output the corresponding codeword $\boldsymbol{b}$, and the algorithm stops; otherwise, go to Step 3.

ML Metrics For Priority-First Search Decoding

We will establish two different metric functions to be used by the priority-first search algorithm. Both metrics will lead to an ML decoding.

- The first metric is

$$f_1(\boldsymbol{b}_{(\ell)}) = g(\boldsymbol{b}_{(\ell)}) + \varphi_1(\boldsymbol{b}_{(\ell)}), \tag{9}$$

  where $\varphi_1(\boldsymbol{b}_{(\ell)}) = 0$ is the all-zero function.

- The second metric is

$$f_2(\boldsymbol{b}_{(\ell)}) = g(\boldsymbol{b}_{(\ell)}) + \varphi_2(\boldsymbol{b}_{(\ell)}), \tag{10}$$

  with $\varphi_2(\boldsymbol{b}_{(\ell)})$ defined later.

The difference

- f_1 can be computed on-the-fly.
- For the evaluation of f_2, one needs to know *all* received symbols, but the computational complexity of f_2 is one order of magnitude less than that of f_1.

Recursive Maximum-Likelihood Metric g

- Let subcode $\mathcal{C}_\theta$ be the set of codewords that satisfy $\mathbb{B}^T\mathbb{B} = \mathbb{G}_\theta$, where θ takes value in Θ. Hence, $\mathcal{C} = \cup_{\theta\in\Theta}\mathcal{C}_\theta$, and $\mathcal{C}_\theta \cap \mathcal{C}_\eta = \emptyset$ whenever $\theta \neq \eta$.
- Since a transmitted codeword belongs to only one of the subcodes, to maintain *individual stacks* for priority-first codeword searching over *each* subcode will introduce considerable unnecessary decoding burden.
- Hence, only one stack is maintained during the entire priority-first search, and the metric function values for different subcodes are compared and sorted in the same stack.
- The path to be expanded next is therefore the one whose metric function value is the smallest globally.

Recursive Maximum-Likelihood Metric g

- By denoting $\mathbb{D}_\theta = \mathbb{G}_\theta^{-1} = (\mathbb{B}^T\mathbb{B})^{-1}$, and letting the matrix entry of $\mathbb{D}_\theta$ be $\delta_{i,j}^{(\theta)}$, we have for $1 \le \ell \le N-1$,

$$g(\boldsymbol{b}_{(\ell+1)}) \;=\; g(\boldsymbol{b}_{(\ell)}) + \max_{\eta\in\Theta} \alpha_{\ell+1}^{(\eta)} - b_{\ell+1} \sum_{i=0}^{P-1}\sum_{j=0}^{P-1} \delta_{i,j}^{(\theta)} \mathrm{Re}\left\{ y_{\ell+i+1} \cdot u_j(\boldsymbol{b}_{(\ell+1)}) \right\},$$

where

$$\alpha_{\ell+1}^{(\eta)} \triangleq \sum_{n=1}^{\ell} |w_{\ell+1,n}^{(\eta)}| + \frac{1}{2}|w_{\ell+1,\ell+1}^{(\eta)}| \tag{11}$$

and for $0 \le j \le P-1$,

$$u_j(\boldsymbol{b}_{(\ell+1)}) = u_j(\boldsymbol{b}_{(\ell)}) + \frac{1}{2}\left(b_\ell y_{\ell+j}^* + b_{\ell+1} y_{\ell+1+j}^* \right).$$

- This shows that we can recursively compute $g(\boldsymbol{b}_{(\ell+1)})$ and $\{u_j(\boldsymbol{b}_{(\ell+1)})\}_{0\le j\le P-1}$ from the previous $g(\boldsymbol{b}_{(\ell)})$ and $\{u_j(\boldsymbol{b}_{(\ell)})\}_{j=0}^{P-1}$ using $y_{\ell+1}, y_{\ell+2}, \ldots, y_{\ell+P}$ and $b_{\ell+1}$, and setting as initial condition $g(\boldsymbol{b}_{(0)}) = u_j(\boldsymbol{b}_{(0)}) = b_0 = 0$ for $0 \le j \le P-1$.

Heuristic Function φ_1

We next derive the first heuristic function that validates (1).

$$\sum_{m=1}^{\ell} \max_{\eta\in\Theta} \alpha_m^{(\eta)} - \frac{1}{2}\sum_{m=1}^{\ell}\sum_{n=1}^{\ell} w_{m,n}^{(\theta)} b_m b_n + \varphi(\boldsymbol{b}_{(\ell)})$$
$$\leq \min_{\{\tilde{\boldsymbol{b}}\in\mathcal{C}\,:\,\tilde{\boldsymbol{b}}_{(\ell)}=\boldsymbol{b}_{(\ell)}\}} \left[\sum_{m=1}^{N} \max_{\eta\in\Theta} \alpha_m^{(\eta)} - \frac{1}{2}\sum_{m=1}^{N}\sum_{n=1}^{N} w_{m,n}^{(\theta)} b_m b_n + \varphi(\tilde{\boldsymbol{b}})\right].$$

Hence, in addition to $\varphi(\tilde{\boldsymbol{b}}) = 0$, the heuristic function should satisfy

$$\varphi(\boldsymbol{b}_{(\ell)}) \leq \sum_{m=\ell+1}^{N} \max_{\eta\in\Theta} \alpha_m^{(\eta)}$$
$$- \max_{\{\tilde{\boldsymbol{b}}\in\mathcal{C}:\,\tilde{\boldsymbol{b}}_{(\ell)}=\boldsymbol{b}_{(\ell)}\}} \left(\sum_{m=\ell+1}^{N} \tilde{b}_m \sum_{n=1}^{\ell} w_{m,n}^{(\theta)} b_n + \frac{1}{2}\sum_{m=\ell+1}^{N}\sum_{n=\ell+1}^{N} w_{m,n}^{(\theta)} \tilde{b}_m \tilde{b}_n\right).$$

It is apparent that the all-zero function is the largest one that satisfies this inequality subject to no dependence on the future route and future receptions, i.e., $\{\tilde{b}_m\}_{m\geq\ell+1}$ and $\{w_{m,n}^{(\theta)}\}_{m\geq\ell+1,n\geq\ell+1}$. Hence, we choose $\varphi_1(\boldsymbol{b}_{(\ell)}) = 0$.

Heuristic Function φ_2

Upon reception of all $y_1, \ldots, y_L$, the heuristic function that satisfies (12) regardless of $\tilde{b}_{\ell+1}, \ldots, \tilde{b}_N$ can be increased to

$$\varphi_2(\boldsymbol{b}_{(\ell)}) \triangleq \sum_{m=\ell+1}^{N} \max_{\eta \in \Theta} \alpha_m^{(\eta)} - \sum_{m=\ell+1}^{N} \left| v_m^{(\theta)}(\boldsymbol{b}_{(\ell)}) \right| - \beta_\ell^{(\theta)}, \qquad (12)$$

where for $1 \leq \ell, m \leq N$ and $\theta \in \Theta$,

$$v_m^{(\theta)}(\boldsymbol{b}_{(\ell)}) \triangleq \sum_{n=1}^{\ell} w_{m,n}^{(\theta)} b_n = v_m^{(\theta)}(\boldsymbol{b}_{(\ell-1)}) + b_\ell w_{\ell,m}^{(\theta)}$$

and

$$\beta_\ell^{(\theta)} = \beta_{\ell-1}^{(\theta)} - \sum_{n=\ell+1}^{N} |w_{\ell,n}^{(\theta)}| - \frac{1}{2}|w_{\ell,\ell}^{(\theta)}|$$

with initial conditions $v_m^{(\theta)}(\boldsymbol{b}_{(0)}) = b_0 = 0$, and $\beta_0^{(\theta)} = \sum_{m=1}^{N} \alpha_m^{(\theta)}$.

Simulation Results

- The channel parameters $\boldsymbol{h}$ used in our simulations is zero-mean complex-Gaussian distributed with

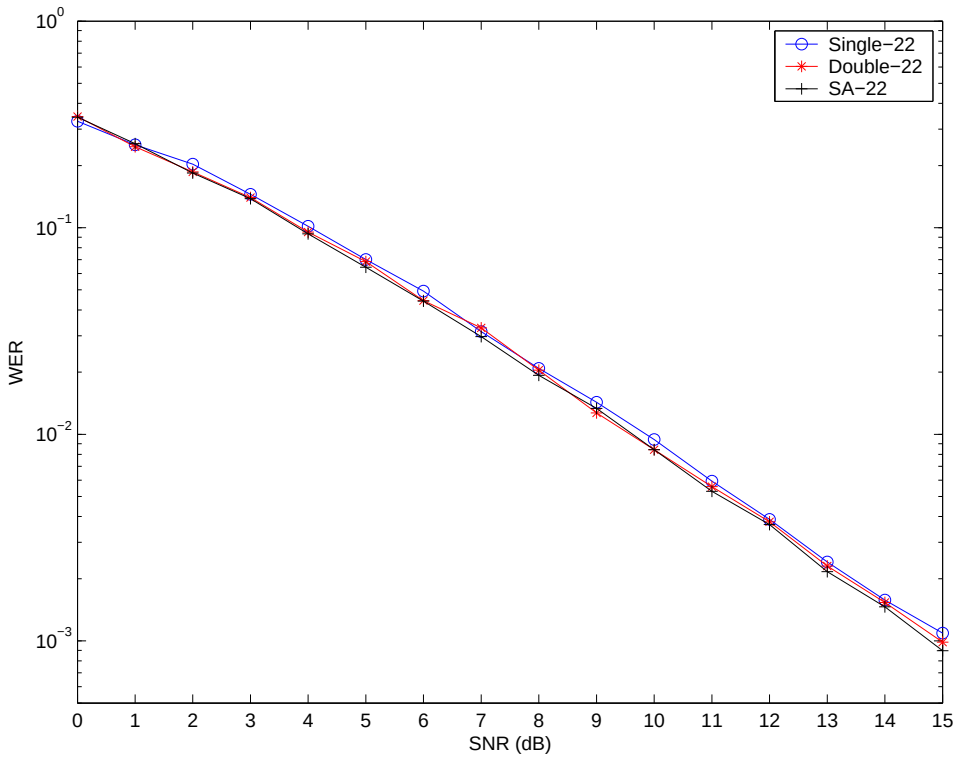
$$E[\boldsymbol{h}\boldsymbol{h}^H] = (1/P)\mathbb{I}_P \quad \text{and} \quad P = 2.$$

- The mapping between the bit patterns and the codewords of the given computer-searched code is obtained through simulated annealing by minimizing the upper bound of

$$\text{BER} \leq \frac{1}{2^K} \sum_{i=1}^{2^K} \sum_{j=1, j\neq i}^{2^K} \frac{d(\boldsymbol{m}_i, \boldsymbol{m}_j)}{K} \Pr\left(\hat{\boldsymbol{b}} = \boldsymbol{b}_j \middle| \boldsymbol{b}_i \text{ transmitted}\right),$$

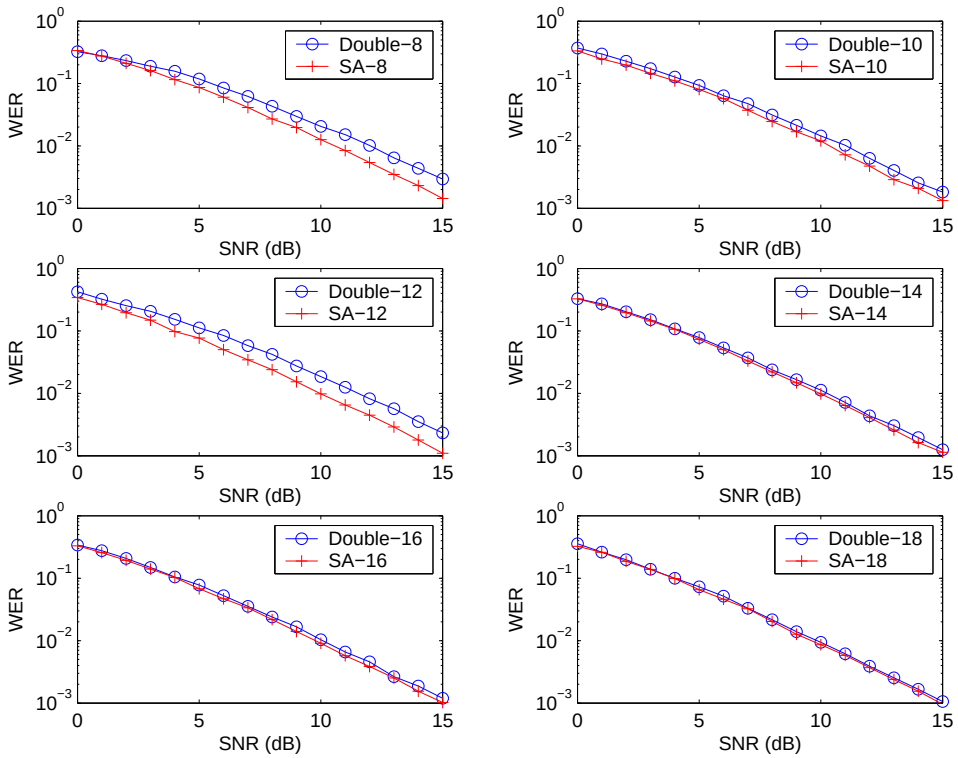
where $\boldsymbol{m}_i$ is the information sequence corresponding to code word $\boldsymbol{b}_i$.

Simulation Results



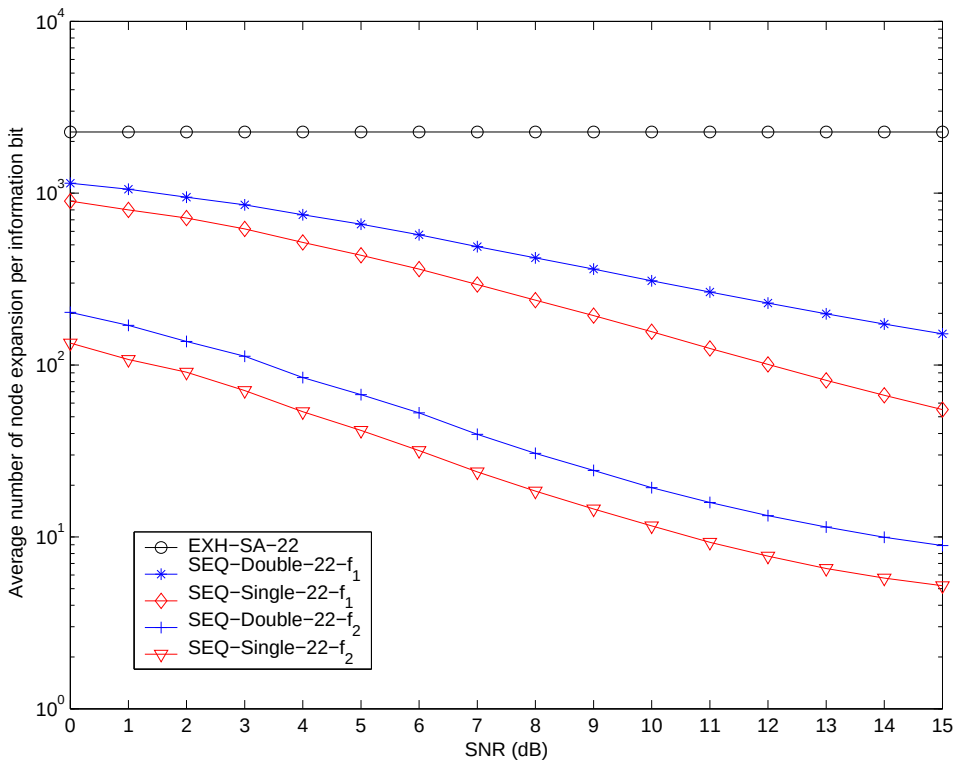
The maximum-likelihood word error rates (WERs) of the computer-searched half-rate code by simulated annealing by Skoglund *et al.* (SA-22), the constructed half-rate code with double code trees (Double-22), and the constructed half-rate code with single code tree (Single-22). The codeword length is $N = 22$.

Simulation Results



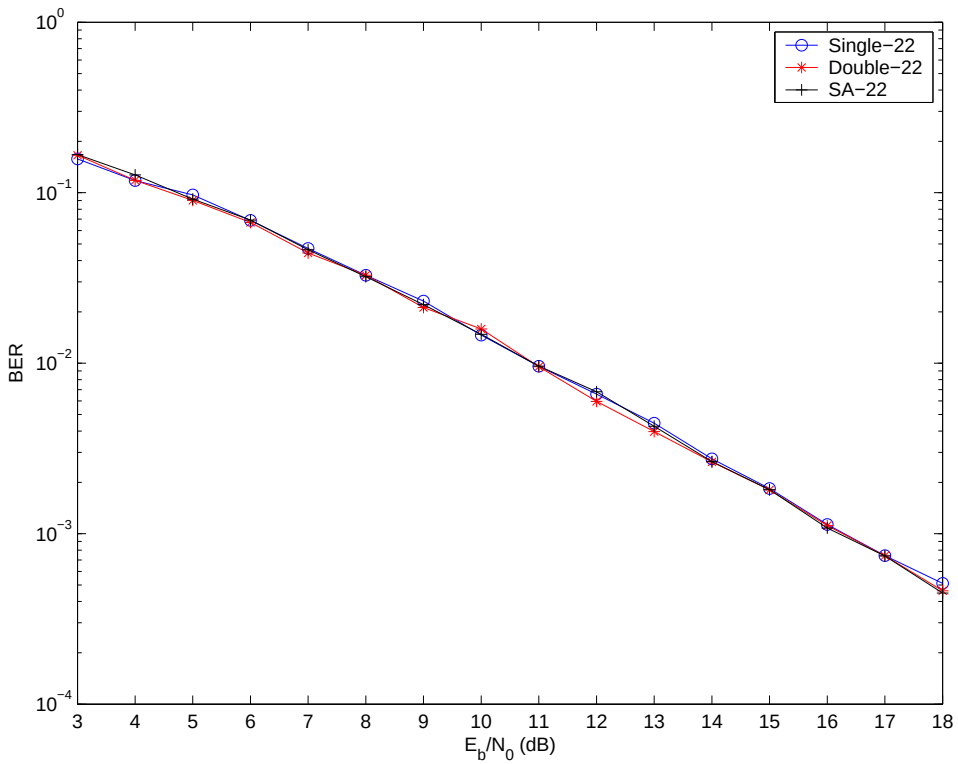
The maximum-likelihood word error rates (WERs) of the computer-searched code by simulated annealing (SA-N) and the constructed half-rate code with double code trees (Double-N).

Simulation Results



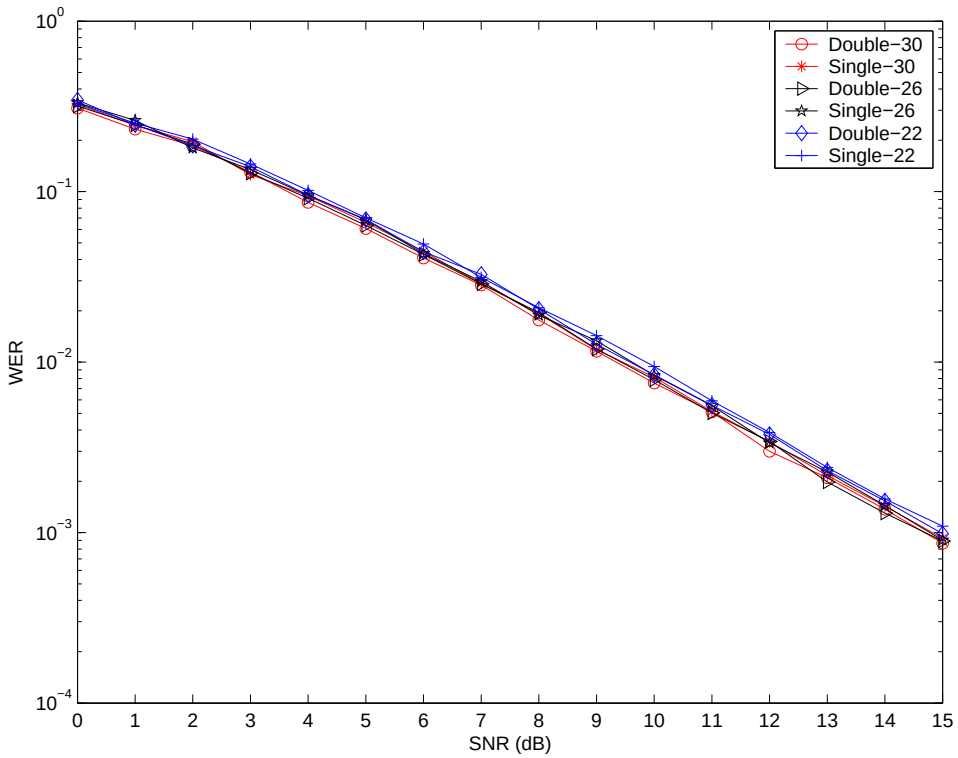
The average numbers of node expansions per information bit for the simulated-annealing-based computer-searched code by Skoglund *et al.* by exhaustive decoding (EXH-SA-22), and the constructed single-tree (SEQ-Single-22) and double-tree (SEQ-Double-22) codes using the priority-first search decoding guided by either metric function f_1 or metric function f_2.

Simulation Results



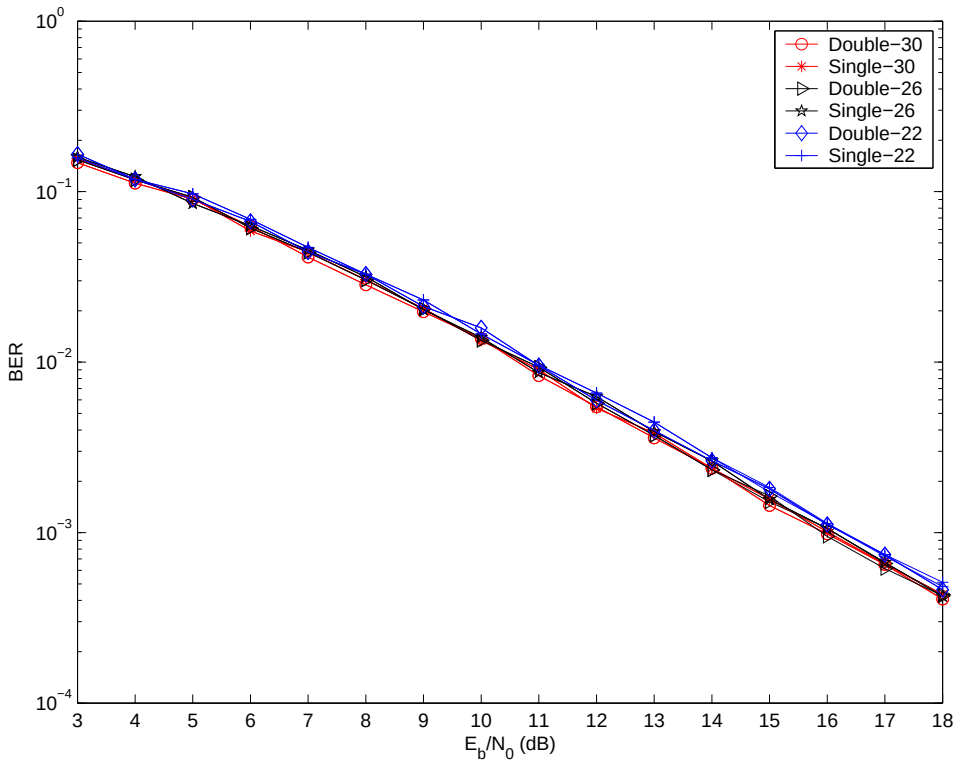
Bit error rates (BERs) for the simulation of codes illustrated in slide 36.

Simulation Results



Word error rates (WERs) for the codes of Single-22, Double-22, Single-26, Double-26, Single-30 and Double-30.

Simulation Results



Bit error rates (BERs) for the codes of Single-22, Double-22, Single-26, Double-26, Single-30 and Double-30.

Simulation Results

Average number of node expansions per information bit for the priority-first search decoding of the constructed half-rate codes of length 22, 26, and 30.

SNR	5dB	6dB	7dB	8dB	9dB	10dB	11dB	12dB	13dB	14dB	15dB	∞ dB
Double-22-f_1	671	590	506	436	375	320	274	236	204	178	156	54
Double-22-f_2	68	55	42	32	26	20	17	14	12	10	9	6
ratio of f_1/f_2	9.8	10.7	12.0	13.6	14.4	16.0	16.1	16.8	17.0	17.8	17.3	9.0
Double-26-f_1	2361	2006	1695	1416	1189	981	813	677	523	499	392	105
Double-26-f_2	175	130	94	69	53	39	29	23	18	15	13	6
ratio of f_1/f_2	13.5	15.4	18.0	20.5	22.4	25.2	28.0	29.4	29.1	33.3	30.2	17.5
Double-30-f_1	8455	7073	5760	5133	3759	3430	2644	1996	1765	1368	1081	192
Double-30-f_2	459	332	232	166	119	86	60	44	33	25	20	7
ratio of f_1/f_2	18.4	21.3	24.8	30.9	31.6	39.9	44.1	45.4	53.4	54.7	54.1	27.4
Single-22-f_1	460	371	308	250	200	163	130	105	85	69	57	12
Single-22-f_2	45	33	26	20	15	12	10	8	7	6	5	4
ratio of f_1/f_2	10.2	11.2	11.8	12.5	13.3	13.5	13.0	13.1	12.1	11.5	11.4	3.0
Single-26-f_1	1635	1328	1061	839	666	522	403	312	244	191	152	21
Single-26-f_2	112	79	57	42	31	23	17	13	11	9	7	4
ratio of f_1/f_2	14.6	16.8	18.6	20.0	21.5	22.7	23.7	23.9	22.2	21.2	21.7	5.3
Single-30-f_1	5871	4695	3857	2924	2335	1813	1328	884	805	572	416	39
Single-30-f_2	284	199	144	101	72	51	35	26	18	14	11	4
ratio of f_1/f_2	20.6	23.6	26.8	29.0	32.4	35.5	38.0	34.0	44.7	40.9	37.8	9.8

The attained diversity levels of codes, which are least-square-approximated based on WER performance curves within 8–15 dBs.

Diversity	$N=8$	$N=10$	$N=12$	$N=14$	$N=16$	$N=18$	$N=22$	$N=26$	$N=30$
SA-N	1.84	1.84	1.88	1.87	1.89	1.87	1.91	—	—
Single-N	—	—	—	—	—	—	1.89	1.90	1.87
Double-N	1.67	1.80	1.79	1.85	1.88	1.87	1.87	1.89	1.87

Summary

- An algorithm to systematically construct codes that allow joint channel estimation and error correction.
- The algorithm is based on the optimal signal-to-noise ratio framework and requires every codeword to satisfy a self-orthogonal property that helps to counter the effects of multi-path fading.
- Simulations demonstrate that the constructed codes have almost identical performance as the best computer-searched codes.
- Two different ML decoding metrics are derived:
  - The first one can be used in an on-the-fly fashion, while the second one that results in a much lower decoding complexity requires the knowledge of all channel outputs.

Codes Designs for Frequency-Selective Varying Fading Channels

Motivation

- In previous chapter, it is assumed that the channel coefficients $\boldsymbol{h}$ are invariant in each coding block of length $L = N + P - 1$.
- In this chapter, we will show that the approaches employed in previous sections can also be readily applicable to the situation that $\boldsymbol{h}$ may change in every Q symbol, where $Q < L$.

System Model

- For $1 \leq k \leq M = \lceil L/Q \rceil$, let $\boldsymbol{h}_k \triangleq [h_{1,k} \;\; h_{2,k} \;\; \cdots \;\; h_{P,k}]^T$ be the constant channel coefficients at the kth sub-block.
- Denote by $\boldsymbol{b}_k = [b_{(k-1)Q-P+2} \;\; \cdots \;\; b_{(k-1)Q+1} \;\; \cdots \;\; b_{kQ}]^T$ the portion of $\boldsymbol{b}$, which will affect the output portion

$$\boldsymbol{y}_k = [y_{(k-1)Q+1} \;\; y_{(k-1)Q+2} \;\; \cdots \;\; y_{kQ}],$$

  where we assume $b_j = 0$ for $j \leq 0$ and $j > N$ for notational convenience.

System Model

- Then, for a channel whose coefficients change in every Q symbol, the system model defined in (2) remains as

$$\boldsymbol{y} = \mathbb{B}\boldsymbol{h} + \boldsymbol{n}$$

except that both $\boldsymbol{y}$ and $\boldsymbol{n}$ extend as $MQ \times 1$ vectors with $y_j = n_j = 0$ for $j > L$, and $\mathbb{B}$ and $\boldsymbol{h}$ have to be re-defined as

$$\mathbb{B} \triangleq \mathbb{B}_1 \oplus \mathbb{B}_2 \oplus \cdots \oplus \mathbb{B}_M \quad \text{and} \quad \boldsymbol{h} \triangleq \begin{bmatrix} \boldsymbol{h}_1^H & \boldsymbol{h}_2^H & \cdots & \boldsymbol{h}_M^H \end{bmatrix}^H,$$

where $\mathbb{B}_k = [\mathbf{0}_{Q\times(P-1)} \quad \mathbb{I}_Q][\boldsymbol{b}_k \quad \tilde{\mathbb{E}}\boldsymbol{b}_k \quad \cdots \quad \tilde{\mathbb{E}}^{P-1}\boldsymbol{b}_k]$ is a $Q \times P$ matrix,

$$\tilde{\mathbb{E}} \triangleq \begin{bmatrix} 0 & 0 & \cdots & 0 & 0 \\ 1 & 0 & \ddots & 0 & 0 \\ 0 & 1 & \ddots & 0 & 0 \\ 0 & 0 & \cdots & 1 & 0 \end{bmatrix}_{(Q+P-1)\times(Q+P-1)}$$

and "$\oplus$" is the direct sum operator of two matrices.

System Model

- Based on the new system model, we have $\mathbb{P}_B = P_{B_1} \oplus P_{B_2} \oplus \cdots \oplus P_{B_M}$, where $\mathbb{P}_{B_k} = \mathbb{B}_k(\mathbb{B}_k^T \mathbb{B}_k)^{-1}\mathbb{B}_k^T$, and Eq. (6) becomes:

$$\hat{\boldsymbol{b}} = \arg\max_{\boldsymbol{b} \in \mathcal{C}} \sum_{k=1}^{M} \left\| \mathrm{vec}(\boldsymbol{y}_k \boldsymbol{y}_k^H) - \mathrm{vec}(\mathbb{P}_{B_k}) \right\|^2 \tag{13}$$

Code Design

Step 1. Fix $b_1 = -1$*, and choose a certain integer* Δ *defined later. Find* 2^K *codewords of the* (N, K) *code by repeating Steps 2–4 for* $0 \le i \le 2^K - 1$*.*

Step 2. Let $\rho_{\min} = 0$ *and* $\rho = i \cdot \Delta$*, where*

$$\Delta = |\mathcal{A}_P(b_1 = -1)|/(2^K - 1).$$

Step 3. For $\ell = 2$ *to* N*, assign the* ℓ*-th bit of the* i*-th codeword,* b_ℓ*, according to that if* $\rho < \rho_{\min} + \gamma_\ell$*, then* $b_\ell = -1$*; else,* $b_\ell = 1$ *and* $\rho_{\min} = \rho_{\min} + \gamma_\ell$*, where*

$$\gamma_\ell = |\mathcal{A}_P(b_1, \ldots, b_{\ell-1}, b_\ell = -1)|,$$

which will be defined shortly.

Step 4. Store the i*th codeword* $\boldsymbol{b}$*, and goto Step 2 for the next codeword until all* 2^K *codewords are selected.*

Code Design

- Since $\mathbb{B}^T\mathbb{B} = (\mathbb{B}_1^T\mathbb{B}_1)\oplus(\mathbb{B}_2^T\mathbb{B}_2)\oplus\cdots\oplus(\mathbb{B}_M^T\mathbb{B}_M)$, the maximization of system SNR can be achieved simply by assigning

$$\mathbb{B}_1^T\mathbb{B}_1 = \mathbb{B}_2^T\mathbb{B}_2 = \cdots = \mathbb{B}_M^T\mathbb{B}_M = Q \cdot \mathbb{I}_Q \tag{14}$$

  regardless of the statistics of $\boldsymbol{h}$ if such assignment is possible.

- Approximation to (14) will have to be taken in the true code design. Then, for channels of memory order 1, i.e., $P = 2$, $\mathcal{A}_P(b_1, \ldots, b_{\ell-1}, b_\ell = -1)$ is the set of all binary ± 1-sequences of length N, whose first bits are assigned as the arguments indicate, and which at the same time satisfy (14).

Optimal Priority-First Search Decoding

- The recursive on-the-fly metric is:

$$g(\boldsymbol{b}_{(\ell)}) - g(\boldsymbol{b}_{(\ell-1)}$$

$$= \begin{cases} \max\limits_{1\le\eta\le\Theta} \alpha_{s,k}^{(\eta)} - b_\ell \sum\limits_{i=0}^{P-1}\sum\limits_{j=0}^{P-1} \delta_{i,j,k}^{(\theta)} \mathrm{Re}\{\tilde{y}_{s+i,k} \cdot u_{j,k}(\boldsymbol{b}_{(\ell)})\}, \text{ for } P \le s \le Q \\ \max\limits_{1\le\eta\le\Theta} \alpha_{r,k}^{(\eta)} + \max\limits_{1\le\eta\le\Theta} \alpha_{s,k+1}^{(\eta)} - b_\ell \sum\limits_{i=0}^{P-1}\sum\limits_{j=0}^{P-1} \Big(\delta_{i,j,k}^{(\theta)} \mathrm{Re}\{\tilde{y}_{r+i,k} \cdot u_{j,k}(\boldsymbol{b}_{(\ell)})\} \\ \qquad\qquad + \delta_{i,j,k+1}^{(\theta)} \mathrm{Re}\{\tilde{y}_{s+i,k+1} \cdot u_{j,k+1}(\boldsymbol{b}_{(\ell)})\} \Big), \text{ otherwise.} \end{cases}$$

where $-P+2 \le \ell \le N$, $s = [(\ell + P - 2) \bmod Q] + 1$, $r = s + Q$, $k = \max\{\lceil \ell/Q \rceil, 1\}$, $\alpha_{s,k}^{(\eta)} \triangleq \sum_{n=1}^{s-1} \left| w_{s,n,k}^{(\eta)} \right| + \frac{1}{2} \left| w_{s,s,k}^{(\eta)} \right|$ and $u_{j,k}(\boldsymbol{b}_{(\ell+1)}) \triangleq u_{j,k}(\boldsymbol{b}_{(\ell)}) + \frac{1}{2}\left(b_\ell \tilde{y}_{s+j,k}^* + b_{\ell+1}\tilde{y}_{s+j+1,k}^*\right)$ with initial values $g(\boldsymbol{b}_{(-P+1)}) = 0$ and $u_{j,k}(\boldsymbol{b}_{((k-1)Q-P+2)}) = 0$ for $0 \le j \le P-1$ and $1 \le k \le M$.

Optimal Priority-First Search Decoding

- In addition, the low-complexity heuristic function is given by:

$$
\varphi_2(\boldsymbol{b}_{(\ell)})
\triangleq
\begin{cases}
\displaystyle\sum_{m=s+1}^{Q+P-1} \max_{1\le\eta\le\Theta} \alpha_{m,k}^{(\eta)} - \sum_{m=s+1}^{Q+P-1} \left| v_{m,k}^{(\theta)}(\boldsymbol{b}_{(\ell)}) \right| - \beta_{s,k}^{(\theta)} \\
\qquad \displaystyle + \sum_{\kappa=k+1}^{M} \left(\sum_{m=1}^{Q+P-1} \max_{1\le\eta\le\theta} \alpha_{m,\kappa}^{(\eta)} - \beta_{0,\kappa}^{(\theta)} \right), & \text{for } P \le s \le Q; \\
\displaystyle\sum_{m=s+1}^{Q+P-1} \max_{1\le\eta\le\Theta} \alpha_{m,k+1}^{(\eta)} - \sum_{m=s+1}^{Q+P-1} \left| v_{m,k+1}^{(\theta)}(\boldsymbol{b}_{(\ell)}) \right| - \beta_{s,k+1}^{(\theta)} \\
\qquad \displaystyle + \sum_{m=r+1}^{Q+P-1} \max_{1\le\eta\le\Theta} \alpha_{m,k}^{(\eta)} - \sum_{m=r+1}^{Q+P-1} \left| v_{m,k}^{(\theta)}(\boldsymbol{b}_{(\ell)}) \right| - \beta_{r,k}^{(\theta)} \\
\qquad \displaystyle + \sum_{\kappa=k+2}^{M} \left(\sum_{m=1}^{Q+P-1} \max_{1\le\eta\le\theta} \alpha_{m,\kappa}^{(\eta)} - \beta_{0,\kappa}^{(\theta)} \right), & \text{otherwise,}
\end{cases}
$$

Optimal Priority-First Search Decoding

where s, r and k are defined the same as for $g(\cdot)$,

$$v_{m,k}^{(\theta)}(\boldsymbol{b}_{(\ell)}) \triangleq \sum_{n=1}^{s} w_{m,n,k}^{(\theta)} b_{(k-1)Q+P+n-1} = v_{m,k}^{(\theta)}(\boldsymbol{b}_{(\ell-1)}) + w_{s,m,k}^{(\theta)} b_{\ell},$$

and

$$\beta_{s,k}^{(\theta)} = \beta_{s-1,k}^{(\theta)} - \sum_{n=s+1}^{Q+P-1} \left| w_{s,n,k}^{(\theta)} \right| - \frac{1}{2} \left| w_{s,s,k}^{(\theta)} \right|$$

with initial values $v_{m,k}^{(\theta)}(\boldsymbol{b}_{(k-1)Q-P+2}) = 0$ and $\beta_{0,k}^{(\theta)} = \sum_{m=1}^{Q+P-1} \alpha_{m,k}^{(\theta)}$.

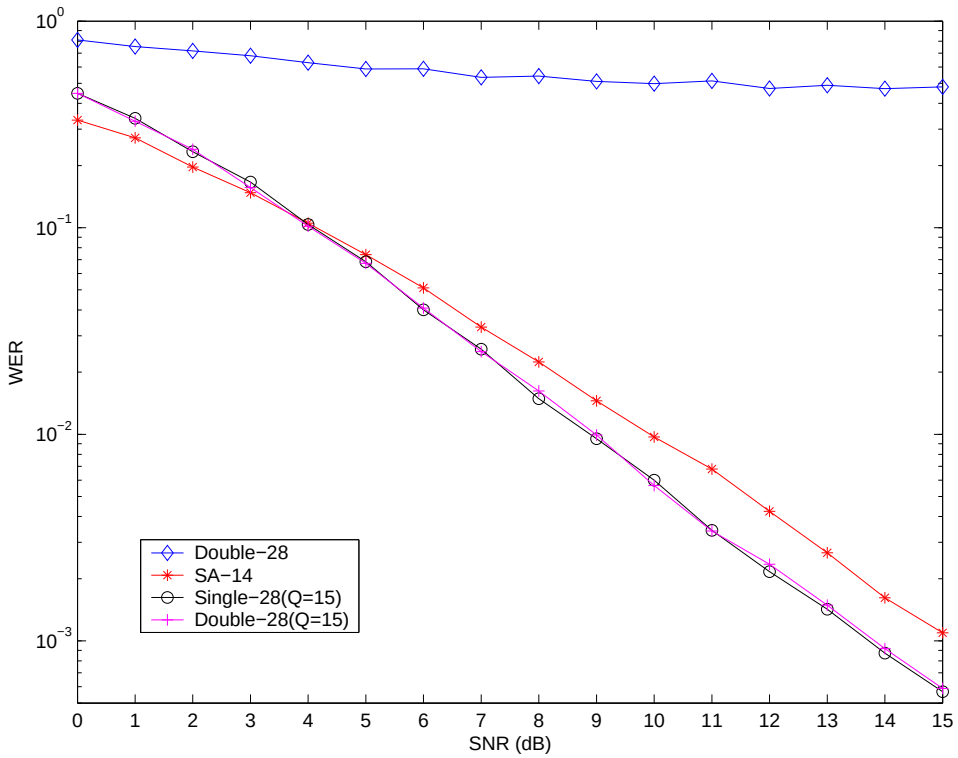
Optimal Priority-First Search Decoding

- If the single-tree code is adopted, $\varphi_2(\cdot)$ can be further reduced to:

$$\varphi_2(\boldsymbol{b}_{(\ell)}) \triangleq \begin{cases} \displaystyle\sum_{m=s+1}^{Q+P-1} \alpha_{m,k}^{(1)} - \sum_{m=s+1}^{Q+P-1} \left| v_{m,k}^{(1)}(\boldsymbol{b}_{(\ell)}) \right| - \beta_{s,k}^{(1)} & \text{for } P \leq s \leq Q; \\ \displaystyle\sum_{m=s+1}^{Q+P-1} \alpha_{m,k+1}^{(1)} - \sum_{m=s+1}^{Q+P-1} \left| v_{m,k+1}^{(1)}(\boldsymbol{b}_{(\ell)}) \right| - \beta_{s,k+1}^{(1)} & \\ \displaystyle\quad + \sum_{m=r+1}^{Q+P-1} \alpha_{m,k}^{(1)} - \sum_{m=r+1}^{Q+P-1} \left| v_{m,k}^{(1)}(\boldsymbol{b}_{(\ell)}) \right| - \beta_{r,k}^{(1)} & \text{otherwise,} \end{cases}$$

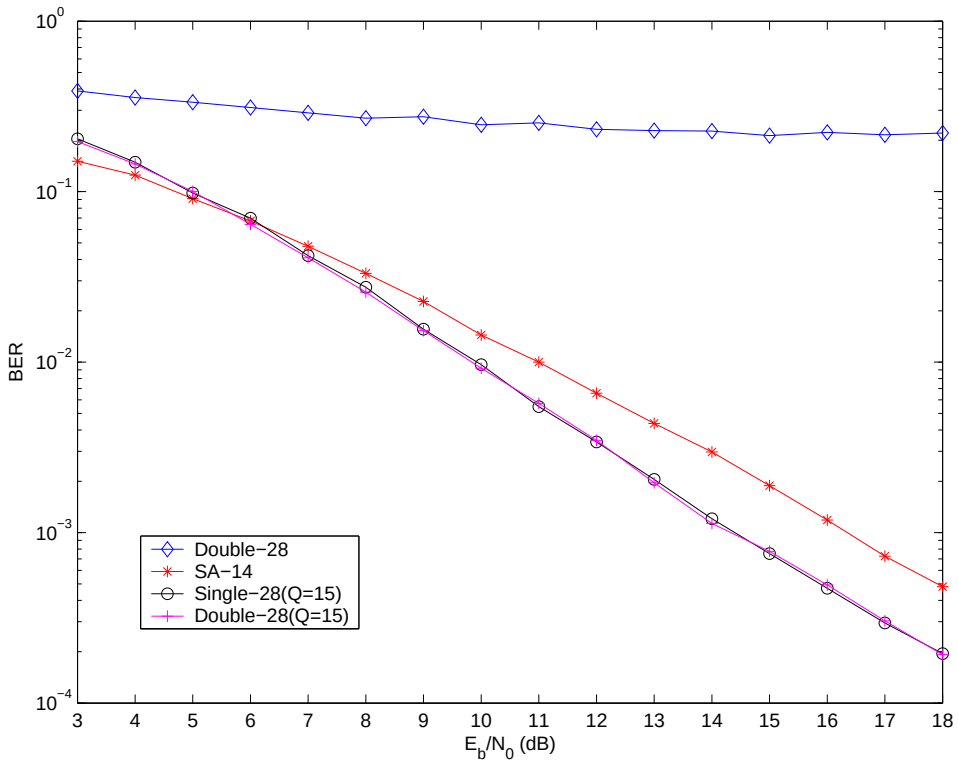
since $\sum_{m=1}^{Q+P-1} \max_{1 \leq \eta \leq \theta} \alpha_{m,\kappa}^{(\eta)} - \beta_{0,\kappa}^{(\theta)} = \sum_{m=1}^{Q+P-1} \alpha_{m,\kappa}^{(1)} - \beta_{0,\kappa}^{(1)} = 0$; hence, a sub-blockwise low-complexity on-the-fly decoding can indeed be conducted under the single code tree condition.

Simulation Results



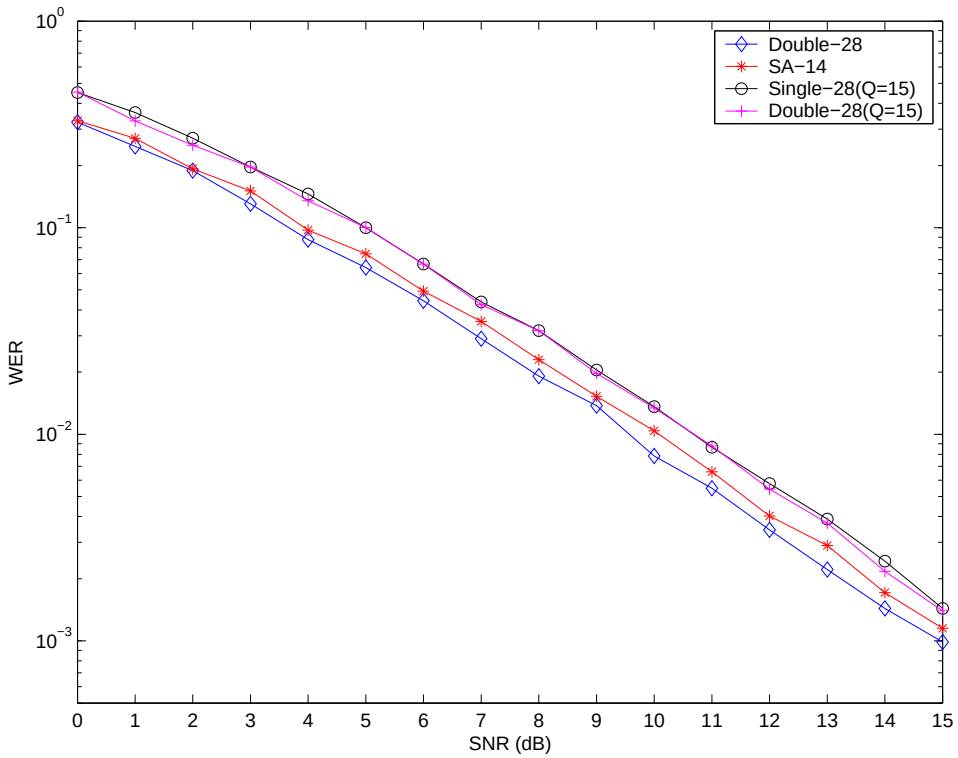
Word error rates (BERs) for the codes of Double-28, SA-14, Single-28(Q=15) and Double-28(Q=15) over the quasi-static channel with $Q_{\text{chan}} = 15$.

Simulation Results



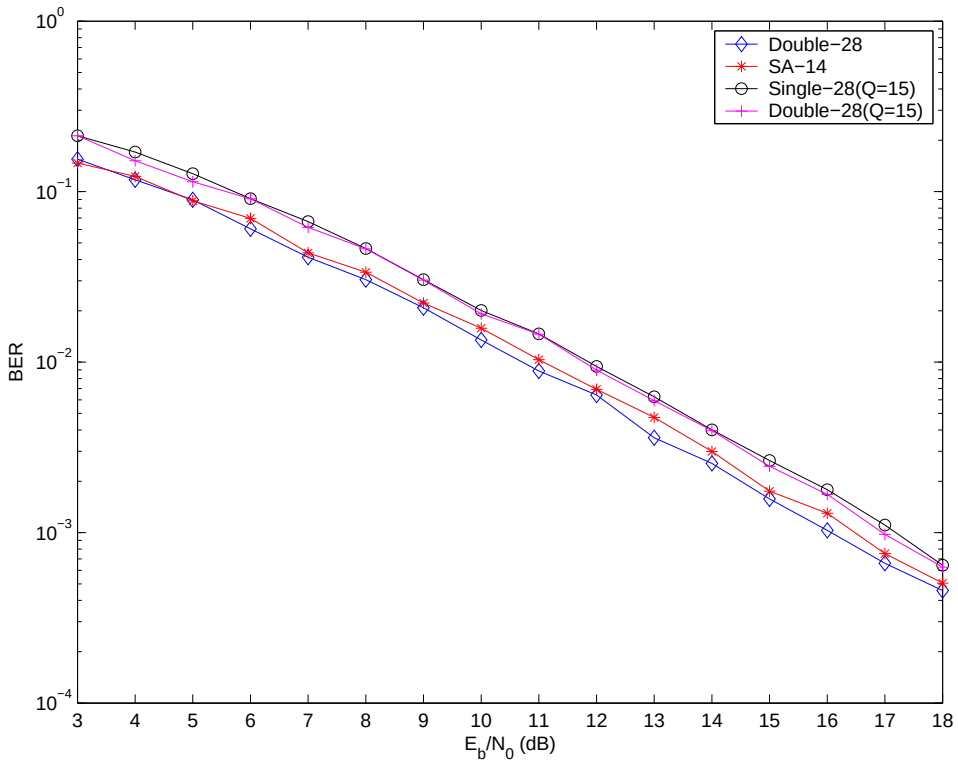
Bit error rates (BERs) for the codes of Double-28, SA-14, Single-28(Q=15) and Double-28(Q=15) over the quasi-static channel with $Q_{\text{chan}} = 15$.

Simulation Results



Word error rates (BERs) for the codes of Double-28, SA-14, Single-28(Q=15) and Double-28(Q=15) over the quasi-static channel with $Q_{\text{chan}} \geq L$.

Simulation Results



Bit error rates (BERs) for the codes of Double-28, SA-14, Single-28(Q=15) and Double-28(Q=15) over the quasi-static channel with $Q_{\text{chan}} \geq L$.

Simulation Results

Average numbers of node expansions per information bit for the codes of Single-28(Q=15) and Double-28(Q=15) using the priority-first search decoding guided by either evaluation function f_1 or evaluation function f_2 over the quasi-static channel with $Q_{\text{chan}} = 15$.

SNR	3dB	4dB	5dB	6dB	7dB	8dB	9dB
Double-28(Q=15)-f_1	2860	2440	2076	1790	1564	1359	1200
Double-28(Q=15)-f_2	1271	1029	877	685	582	484	413
ratio of f_1/f_2	2.3	2.4	2.4	2.6	2.7	2.8	2.9
Single-28(Q=15)-f_1	1658	1367	1074	899	701	593	488
Single-28(Q=15)-f_2	766	625	482	392	321	254	219
ratio of f_1/f_2	2.2	2.2	2.2	2.3	2.2	2.3	2.2

SNR	10dB	11dB	12dB	13dB	14dB	15dB
Double-28(Q=15)-f_1	1040	958	899	811	780	723
Double-28(Q=15)-f_2	353	312	277	250	229	207
ratio of f_1/f_2	2.9	3.1	3.2	3.2	3.4	3.5
Single-28(Q=15)-f_1	448	356	309	277	244	232
Single-28(Q=15)-f_2	177	149	133	121	104	92
ratio of f_1/f_2	2.5	2.4	2.3	2.3	2.3	2.5

Summary

- An extension of the code design for combined channel estimation and error correction to channels with independently varying fading subblocks.
- This design can directly construct a code of any desired code length and code rate, of which the performance is shown to be comparable to the best computer-searched code for the channels simulated.
- Although we only derive the coding scheme and its decoding metric for a fixed Q, further extension to the situation that the channel coefficients $\boldsymbol{h}$ vary nonstationarily as the periods Q_1, Q_2, ..., Q_M are not equal is straightforward.
- Such design may be suitable for, e.g., the frequency-hopping scheme of Global System for Mobile communications (GSM) and Universal Mobile Telecommunications System (UMTS), or the time-hopping scheme in IS-54, in which cases the channel coefficients change (or hop) at protocol-aware scheduled time.

Code Designs for Frequency-Nonselective Varying Fading Channels

Review

- The error correcting code design that jointly considers channel estimation is especially useful in situation when either the fading is rapid enough to preclude a good estimate of channel taps or the cost of implementing the channel estimators is high.
- One example is the reliable delivery of often short-in-length control signal such as channel quality indicator (CQI) in a highly mobile environment.
- At this background, Xu *et al.* proposed a novel nonlinear coding scheme suitable for blind noncoherent detection of the transmitted control signal to the 802.16m standard body.
- In the proposal, the uplink CQI information is encoded using a $(12, 6)$ code. The codeword will then be repeatedly transmitted three times (perhaps through different OFDM channels) in order to further benefit from diversity gain (which can be equivalently regarded as a $(36, 6)$ coding scheme).

Review

- Based on self-orthogonality framework, we propose a *systematic* (N, K) coding scheme that can deal with any given N and K for channels with possibly varying channel coefficients in a coding block.
- Simulations show that our constructed $(36, 6)$ code has almost the same performance as Xu's three-times-repetitive $(12, 6)$ code when the channel independently varies its coefficients three times in a coding block. In case the channel remains constant during the entire coding block, our constructed code has 0.7 dB performance improvement over Xu's code.
- Xu's code is specifically designed for a frequency-nonselective OFDM system, while our systematic code construction scheme can also be applied in a frequency selective environment.
- Our simulation results indicate that with a proper design, a blind-detectable noncohrent code can be made robust for channels whose taps may vary more often than a coding block.

System Model

- Suppose that a codeword $\boldsymbol{b} = [b_1 \ \cdots \ b_N]^T$ is transmitted over a block fading channel of memory order 0, of which channel coefficients may vary in every Q symbols, where $b_i \in \{\pm 1\}$ and $Q > P$.
- It can be derived that the joint maximum-likelihood decoder upon the reception of $\boldsymbol{y}$:

$$\begin{aligned}\hat{\boldsymbol{b}} = \arg\min_{\boldsymbol{b}\in\mathcal{C}} \min_{\boldsymbol{h}} \|\boldsymbol{y} - \mathbb{B}\boldsymbol{h}\|^2 &= \arg\max_{\boldsymbol{b}\in\mathcal{C}} \sum_{k=1}^{M} \left\|\boldsymbol{y}_k\boldsymbol{y}_k^H - \mathbb{P}_{B_k}\right\|^2 \\ &= \arg\max_{\boldsymbol{b}\in\mathcal{C}} \sum_{k=1}^{M} \|\boldsymbol{y}_k^H\boldsymbol{b}_k\|^2,\end{aligned}$$

  where $\boldsymbol{y}_k \triangleq [y_{(k-1)Q+1} \ \ y_{(k-1)Q+2} \ \cdots \ y_{kQ}]$ is the output portion affected by $\boldsymbol{b}_k$, and $\mathbb{P}_{B_k} \triangleq \mathbb{B}_k(\mathbb{B}_k^T\mathbb{B}_k)^{-1}\mathbb{B}_k^T$.
- In the above derivation, we assume that the receiver, although it knows nothing about $\boldsymbol{h}$, has perfect knowledge about the values of Q.

Code Design

- Now, as far as the code design for frequency nonselective channels is concerned, $\mathcal{A}_0(b_1, \ldots, b_\ell)$ is simply the set of all binary ± 1-sequences of length N, whose first ℓ bits are assigned as the arguments indicate, and which at the same time satisfies that

$$\begin{cases} \mathbb{B}_k^T \mathbb{B}_k = Q \text{ for } 1 \leq k < M = \lceil L/Q \rceil \\ \mathbb{B}_M^T \mathbb{B}_M = N - (M-1)Q. \end{cases} \tag{15}$$

- It may not be practical to examine the minimum pairwise distance for all 2^K codewords for the determination of the best Δ.

- Instead, we choose K codewords as representatives. These representative codewords correspond to $\rho = 2^j \Delta$ for $0 \leq j \leq K-1$. We then adopt the Δ that minimizes the pairwise distance among these K codewords.

Code Design

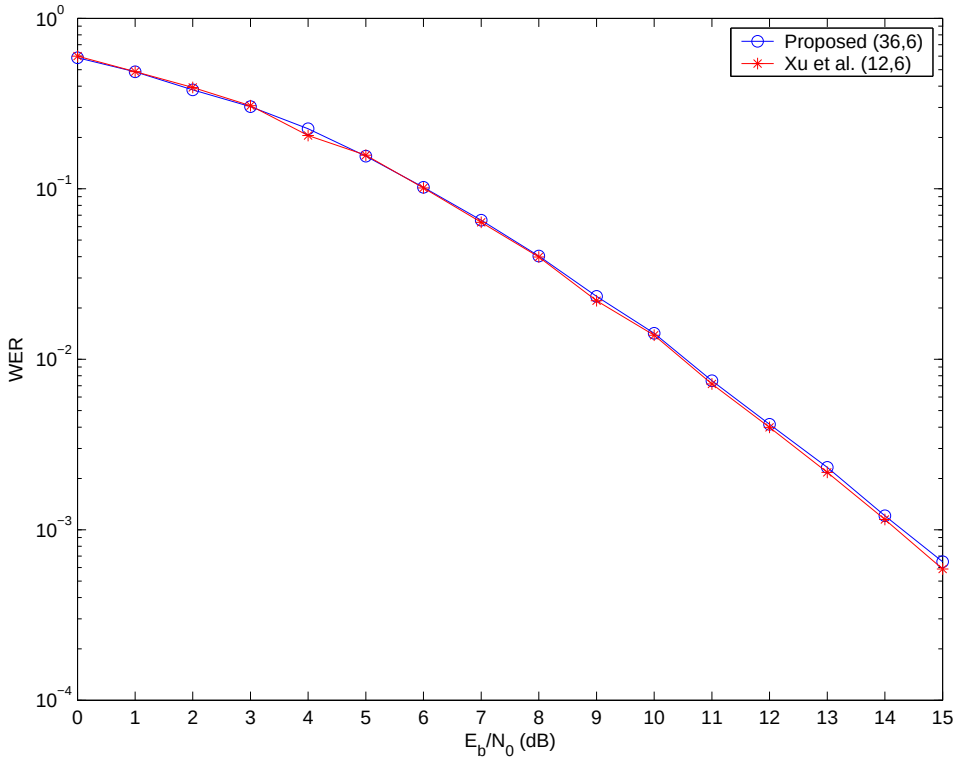
- When $N > K+4$ and $P = 0$, the proposed process of determining Δ is indeed equivalent to that the Δ-th codeword must be of the form

$$[\underbrace{-1 \cdots -1}_{K+1}\ 1\ 1\ \boldsymbol{u}\ 1],$$

where $\boldsymbol{u}$ is a maximum-length shift-register sequence. In other words, the first $K+3$ bits are fixed as $[-1\ \cdots\ -1\ \ 1\ \ 1]$, and the last bit is always equal to 1. This is because under $P = 0$, all binary ± 1-sequences satisfy (15), which results in that $(2^{j+1}\Delta)$-th codeword is exactly the logical left-shift of $(2^j\Delta)$-th codeword.

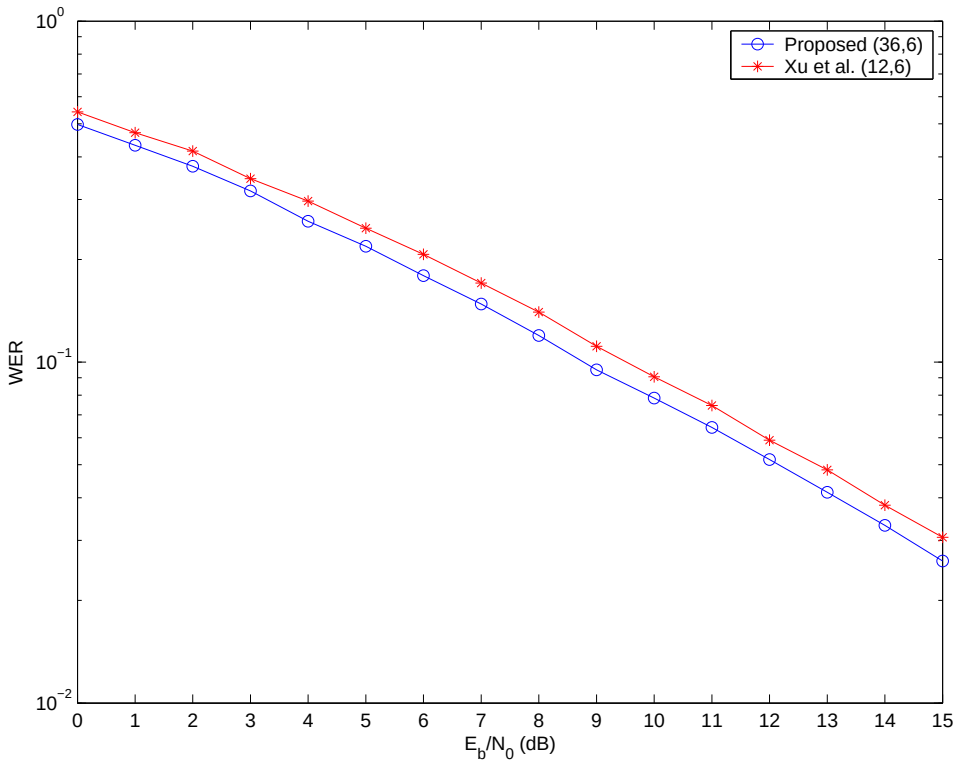
Simulation Results

The channel parameters $\boldsymbol{h}$ is zero-mean complex-Gaussian distributed.



Word error rates (WERs) for the constructed $(36, 6)$ code and Xu's three-times-repetitive $(12, 6)$ code over flat fading channel with coefficients varying independently in every 12 symbols.

Simulation Results



Word error rates (WERs) for the constructed $(36, 6)$ code and Xu's three-times-repetitive $(12, 6)$ code over flat fading channel with coefficients unchanged during the transmission of a codeword.

Summary

- In this chapter, we propose a novel systematic code construction scheme for frequency-nonselective fading channel.
- We then compare our codes with the three-times-repetitive $(12, 6)$ code proposed by Xu *et al.* for use of channel quality indicator (CQI) in uplink control for IEEE 802.16m.
- Simulations show that our constructed $(36, 6)$ code has comparable performance to Xu's code when channel coefficients changes randomly in every 12 symbols.
- If the channel taps remain constant in the entire coding block of length 36, our code outperforms Xu's code by 0.7 dB. This indicates that the new constructed code adapts more robustly to the two simulated scenarios.

A Systematic Space-Time Code Design

Introduction

Categories for coding schemes designed for noncoherent receivers in MIMO flat-fading channel

- The space-time *constellations* based on computer search.
  - Possible obstacles
    - High design complexity in general
    - Need to be redesigned when design assumptions change
    - Requirement of exhaustive search decoding
  - To alleviate these problems, we focus on a systematic code construction and its low-complexity decoder in this work.
  - Consider only **two-transmit-antenna** and **half-rate** code.
- Schemes that couple known space-time block codes with blind detection.

Some notable publications that deal with a similar problem, but cannot be classified into any of the previous two categories.

Examples:

- The so-called *training codes* that incorporate training symbols into their codewords.
  - Such designs are different from our design, which aim at combining channel estimation and error correction.
- Construct a noncoherent code through a mapping from coherent code.
  - Only suboptimal efficient decoder is provided.

System model

A MIMO system with A_{T} transmit antennas and A_{R} receive antennas can be modelled as follows.

$$\begin{bmatrix} y_{1,1} & \cdots & y_{1,A_{\mathrm{R}}} \\ \vdots & \vdots & \vdots \\ y_{N,1} & \cdots & y_{N,A_{\mathrm{R}}} \end{bmatrix} = \begin{bmatrix} b_{1,1} & \cdots & b_{1,A_{\mathrm{T}}} \\ \vdots & \vdots & \vdots \\ b_{N,1} & \cdots & b_{N,A_{\mathrm{T}}} \end{bmatrix} \begin{bmatrix} h_{1,1} & \cdots & h_{1,A_{\mathrm{R}}} \\ \vdots & \vdots & \vdots \\ h_{A_{\mathrm{T}},1} & \cdots & h_{A_{\mathrm{T}},A_{\mathrm{R}}} \end{bmatrix} + \begin{bmatrix} n_{1,1} & \cdots & n_{1,A_{\mathrm{R}}} \\ \vdots & \vdots & \vdots \\ n_{N,1} & \cdots & n_{N,A_{\mathrm{R}}} \end{bmatrix}$$

$$\Downarrow$$

$$\begin{bmatrix} \boldsymbol{y}_1 & \cdots & \boldsymbol{y}_{A_{\mathrm{R}}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{b}_1 & \cdots & \boldsymbol{b}_{A_{\mathrm{T}}} \end{bmatrix} \mathbb{H} + \begin{bmatrix} \boldsymbol{n}_1 & \cdots & \boldsymbol{n}_{A_{\mathrm{R}}} \end{bmatrix},$$

$$\Downarrow$$

$$\mathbb{Y} = \mathbb{B}\mathbb{H} + \mathbb{N},$$

where
$$\begin{cases} b_{i,n} \in \{\pm 1/\sqrt{A_{\mathrm{T}}}\}, \\ \boldsymbol{n}_i \text{ is zero-mean complex Gaussian with } E[\boldsymbol{n}_i \boldsymbol{n}_i^H] = \sigma_n^2 \mathbb{I}_N, \\ \mathbb{I}_N \text{ is the } N \times N \text{ identity matrix,} \\ \text{and } \mathbb{H} \text{ is an unknown constant matrix.} \end{cases}$$

Assumption

- Both transmitter and receiver know nothing about the channel coefficients $\mathbb{H}$.

Decoding Criterion

- The least-square estimate of $\mathbb{H}$ for a given codeword $\mathbb{B}$ is:

$$\hat{\mathbb{H}} \triangleq (\mathbb{B}^T\mathbb{B})^{-1}\mathbb{B}^T\mathbb{Y}$$

- For a given (N, K) block code $\mathcal{C}$, the maximum-likelihood (ML) decision is made based on the well-known generalized likelihood ratio test (GLRT) as

$$\begin{aligned}\hat{\mathbb{B}} &= \arg\min_{\mathbb{B}} \min_{\mathbb{H}} \|\mathbb{Y} - \mathbb{B}\mathbb{H}\|^2 \\ &= \arg\min_{\mathbb{B}} \|\mathbb{Y} - \mathbb{B}\hat{\mathbb{H}}\|^2 \\ &= \arg\min_{\mathbb{B}} \|(\mathbb{I}_N - \mathbb{P}_B)\mathbb{Y}\|^2,\end{aligned}$$

where $\mathbb{P}_B \triangleq \mathbb{B}(\mathbb{B}^T\mathbb{B})^{-1}\mathbb{B}^T$.

Criteria for Good Codes

Several criteria for good codes have been proposed in the literature. We will in particular consider two of them: pairwise full-rank and unitary.

Why pairwise full-rank?

- For $N \geq 2A_{\mathrm{T}}$, $\mathbb{B}(i)$ and $\mathbb{B}(j)$ are *pair-wisely full-rank* if
$$\operatorname{rank}\left(\left[\mathbb{B}(i)\ \ \mathbb{B}(j)\right]\right) = 2A_{\mathrm{T}}.$$
- At high SNR, the pair-wise word error is well approximated by
$$\Pr\left(\hat{\mathbb{B}} = \mathbb{B}(j) \middle| \mathbb{B}(i) \text{ transmitted}\right) \approx \mathcal{Q}\left(\frac{1}{\sqrt{2}}\|\mathbb{H}\|\sqrt{\lambda_{\min}(\mathbb{L}_{ij})}\right)$$
where $\begin{cases} \mathbb{L}_{ij} \triangleq \mathbb{I}_{A_{\mathrm{R}}} \otimes \left(\mathbb{B}(i)^T\left(\mathbb{I}_N - \mathbb{P}_{B(j)}\right)\mathbb{B}(i)\right), \\ \lambda_{\min}(\mathbb{L}_{ij}) \text{ is the smallest eigenvalue of } \mathbb{L}_{ij}, \\ \text{and “}\otimes\text{” indicates the Kronecker product.} \end{cases}$
- If $\operatorname{rank}\left([\mathbb{B}(i)\ \mathbb{B}(j)]\right) \neq 2A_{\mathrm{T}}$, $\det\left|\mathbb{B}(i)^T\left(\mathbb{I}_N - \mathbb{P}_{B(j)}\right)\mathbb{B}(i)\right| = 0$. This implies that $\lambda_{\min}(\mathbb{L}_{ij}) = 0$, and the error probability will be close to $1/2$!

Criteria for Good Codes

Why unitary, i.e., $\mathbb{B}^T\mathbb{B} = (N/A_{\mathrm{T}}) \cdot \mathbb{I}_{A_{\mathrm{T}}}$?

- Unitary codewords can maximize the average signal-to-noise ratio (SNR) regardless of the statistics on $\mathbb{H}$.
- A unitary signal maximizes the capacity when $\mathbb{H}$ is zero-mean complex Gaussian distributed.
- Unitary codewords minimize the union bound of word error rate (WER) at high SNR.

Remark: Therefore, a code that satisfies both pair-wise full-rank and unitary properties necessarily guarantee a good pairwise-error-based union bound, which in turn hints at good performance.

This argument will be confirmed by the subsequent simulations.

The Proposed Code Design

The (N, K) code proposed is $\mathbb{B} = \frac{1}{\sqrt{A_{\mathrm{T}}}} \begin{bmatrix} \boldsymbol{k} & \boldsymbol{k} \odot \boldsymbol{s} \\ -\boldsymbol{k} \odot \boldsymbol{s} & \boldsymbol{k} \end{bmatrix}$

where $\begin{cases} \boldsymbol{k} \in \{\pm 1\}^K \text{ is the information sequence,} \\ \text{“}\odot\text{” denotes the Hadamard product,} \\ \text{if } k_1 = -1, \boldsymbol{s} = \begin{bmatrix} \mathbf{1}_{K-\lceil K/2 \rceil} \\ -\mathbf{1}_{\lceil K/2 \rceil} \end{bmatrix}; \text{Otherwise, } \boldsymbol{s} = \begin{bmatrix} \mathbf{1}_{K-\lceil K/2 \rceil} \\ -\mathbf{1}_{\lceil K/2 \rceil} \end{bmatrix} \\ \mathbf{1}_k \text{ represents a } k \times 1 \text{ all-one vector,} \\ \text{and } \boldsymbol{d} \triangleq [(-1)^0 \ (-1)^1 \ \ldots \ (-1)^{K-1}]^T \end{cases}$

Remarks:

- Our design can be viewed as a high-dimensional variation of Alamouti codes.
- The unitary property is satisfied by the Alamouti code structure.
- By introducing the additional Hadamard product, our code can fulfill the pairwise full-rank as shown in the next two slides.

Full-Rank For Our Design

- For the validity of the **pair-wise full-rank** criterion, it suffices to prove that

$$\det \left| \mathbb{I}_{A_{\mathrm{T}}} - \frac{1}{(N/A_{\mathrm{T}})^2} \mathbb{A}_{i,j} \mathbb{A}_{i,j}^T \right| \neq 0, \tag{16}$$

  for $1 \leq i, j \leq 2^K$ with $i \neq j$, where $\mathbb{A}_{i,j} \triangleq \mathbb{B}(i)^T \mathbb{B}(j)$.

- By denoting respectively the kth eigenvalue and kth eigenvector of $\mathbb{A}_{i,j}\mathbb{A}_{i,j}^T$ by λ_k and $\boldsymbol{u}_k$, the validity of (16) can be verified by showing that $\lambda_k \neq (N/A_{\mathrm{T}})^2$ for every k because

$$\begin{aligned}
(16) \;\Leftrightarrow\; & \det \left| \mathbb{I}_{A_{\mathrm{T}}} - \frac{1}{(N/A_{\mathrm{T}})^2} \sum_{k=1}^{A_{\mathrm{T}}} \lambda_k \boldsymbol{u}_k \boldsymbol{u}_k^T \right| \neq 0 \\
\Leftrightarrow\; & \det \left| \sum_{k=1}^{A_{\mathrm{T}}} \left(1 - \frac{\lambda_k}{(N/A_{\mathrm{T}})^2} \right) \boldsymbol{u}_k \boldsymbol{u}_k^T \right| \neq 0.
\end{aligned}$$

Full-Rank For Our Design

For $N > 2$, we prove that $\lambda_1 \neq (N/A_{\mathrm{T}})^2$ and $\lambda_2 \neq (N/A_{\mathrm{T}})^2$ by the following two cases.

- *Case 1:* $\boldsymbol{s}_i = \boldsymbol{s}_j = \boldsymbol{s}$, where $(\boldsymbol{s}_i, \boldsymbol{k}_i)$ and $(\boldsymbol{s}_j, \boldsymbol{k}_j)$ respectively define code words $\mathbb{B}(i)$ and $\mathbb{B}(j)$. Then,

$$\mathbb{A}_{i,j}\mathbb{A}_{i,j}^T = \mathbb{B}(i)^T\mathbb{B}(j)\left[\mathbb{B}(i)^T\mathbb{B}(j)\right]^T = \begin{bmatrix} (\boldsymbol{k}_i^T\boldsymbol{k}_j)^2 & 0 \\ 0 & (\boldsymbol{k}_i^T\boldsymbol{k}_j)^2 \end{bmatrix}$$

  So, $\lambda_1 = \lambda_2 = (\boldsymbol{k}_i^T\boldsymbol{k}_j)^2 < (N/A_{\mathrm{T}})^2$.

- *Case 2:* $\boldsymbol{s}_i \neq \boldsymbol{s}_j$. Then,

$$\mathbb{A}_{i,j}\mathbb{A}_{i,j}^T = \begin{bmatrix} c & 0 \\ 0 & c \end{bmatrix},$$

  where $\begin{cases} c \triangleq (1/A_{\mathrm{T}})^2(|\boldsymbol{k}_i^T(\boldsymbol{k}_j + \boldsymbol{k}_j \odot \boldsymbol{d})|^2 + |\boldsymbol{k}_i^T(\boldsymbol{c}_{j,j} - \boldsymbol{c}_{j,i})|^2), \\ \text{and } \boldsymbol{c}_{j,i} \triangleq \boldsymbol{k}_j \odot \boldsymbol{s}_i. \end{cases}$

  Accordingly, $\lambda_1 = \lambda_2 = c < (N/A_{\mathrm{T}})^2$.

Priority-First Search Decoding

- Next, the maximum-likelihood priority-first search decoding algorithm for our code is designed.
- It requires a **recursive** decoding metric.
- Further derivation gives an equivalent recursive decoding metric f to the previous criterion:

$$f(\boldsymbol{k}_{(\ell)}) = g(\boldsymbol{k}_{(\ell)}) - \gamma(\boldsymbol{k}_{(\ell)}),$$

where

$$\begin{cases} \boldsymbol{k}_{(\ell)} = [k_1 \; k_2 \; \ldots \; k_\ell]^T, \\ g(\boldsymbol{k}_{(\ell+1)}) = g(\boldsymbol{k}_{(\ell)}) - \beta(\boldsymbol{k}_{(\ell+1)}), \\ \gamma(\boldsymbol{k}_{(\ell)}) = -\displaystyle\sum_{m=\ell+1}^{K} \sum_{n=\ell+1}^{m} \left(|d_{m,n}| \, \mathbf{1}\{s_m = s_n\} + |e_{m,n}| \, \mathbf{1}\{s_m \neq s_n\} \right) \\ \qquad + \dfrac{\sqrt{A_{\mathrm{T}}}}{N} \displaystyle\sum_{m=\ell+1}^{K} \left| \sum_{r=1}^{A_{\mathrm{R}}} \mathbf{Re} \left\{ \sum_{t=0}^{1} y_{m+tK,r} \sum_{i=0}^{1} \sum_{j=0}^{1} (-1)^p s_m^q \cdot u_{i,j}^{(r)}(\boldsymbol{k}_{(\ell)}) \right\} \right|, \end{cases}$$

and

$$\begin{cases} \beta(\boldsymbol{k}_{(\ell+1)}) = k_{\ell+1}\dfrac{\sqrt{A_{\mathrm{T}}}}{N}\displaystyle\sum_{r=1}^{A_{\mathrm{R}}}\mathbf{Re}\left\{\sum_{t=0}^{1} y_{\ell+1+tK,r}\sum_{i=0}^{1}\sum_{j=0}^{1}(-1)^p s_{\ell+1}^q \cdot u_{i,j}^{(r)}(\boldsymbol{k}_{(\ell)})\right\}, \\ p = \lfloor t + (-1)^t(i+j)/2 \rfloor, \\ q = t + |i-j|(-1)^t, \\ u_{i,j}^{(r)}(\boldsymbol{k}_{(\ell+1)}) = u_{i,j}^{(r)}(\boldsymbol{k}_{(\ell)}) + \dfrac{1}{\sqrt{A_{\mathrm{T}}}}k_{\ell+1}s_{\ell+1}^i y_{\ell+1+jK,r}^*, \\ \text{and } \mathbf{1}\{\cdot\} \text{ denotes the set indicator function.} \end{cases}$$

- Since the metric proposed is nondecreasing along every path in the code tree, the maximum-likelihood decoding result is guaranteed.

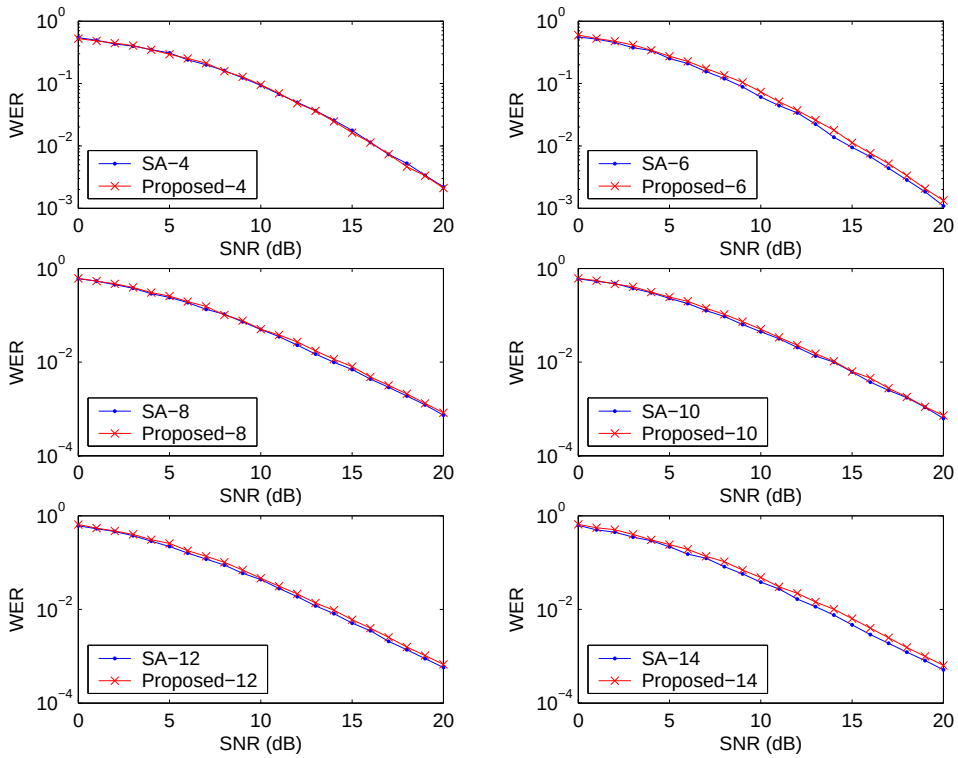
Simulation Results

For simplicity, we take $A_{\mathrm{R}} = 1$ in our simulations, and assume that $\mathbb{H}$ is zero-mean complex Gaussian with $E[\mathbb{H}\mathbb{H}^H] = (1/A_{\mathrm{T}})\mathbb{I}_{A_{\mathrm{T}}}$.

In this setting, the average SNR is give by

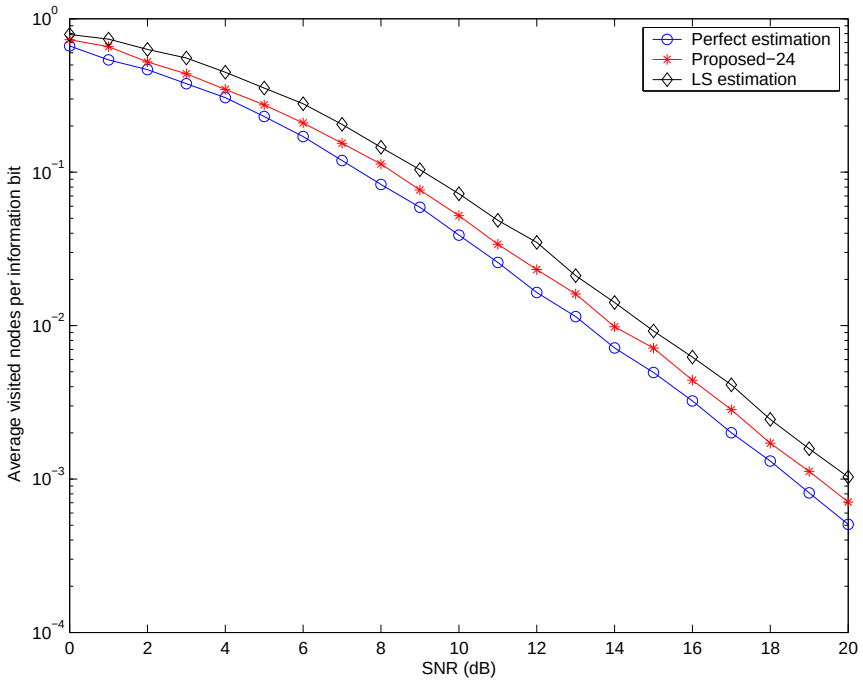
$$\mathrm{SNR} = \frac{E[\mathbb{H}^H\mathbb{H}]}{(K/N)\sigma^2} = \frac{2}{\sigma^2}.$$

Simulation Results



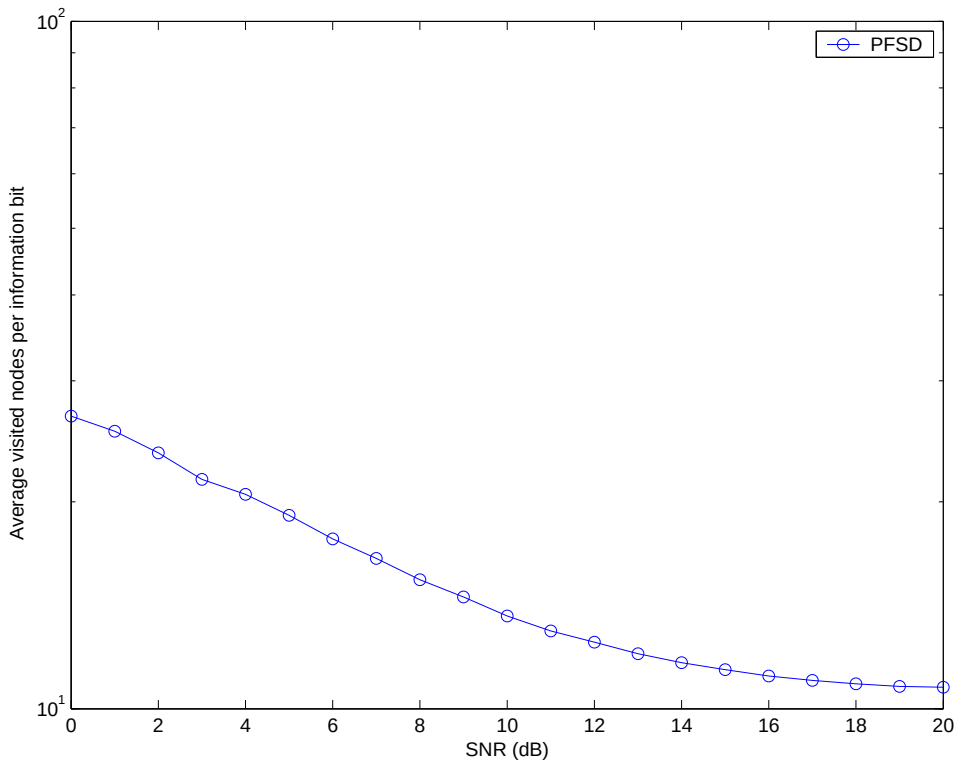
- **Proposed-N**: The proposed codes of block length N.
- **SA-N**: The codes obtained from simulated annealing search according to the criterion of minimizing the union bound.

Simulation Results



- **Proposed-**24: The proposed codes of block length 24.
- **Reference scheme**: The system using a $(17, 12)$ nonlinear code in combination with the Alamouti code mapping and a 7-bit training sequence.

Simulation Results



- The decoding complexity of the priority-first search decoder (PFSD) for the constructed code of length $N = 24$.
- **The (equivalent) complexity of exhaustive decoding is equal to** 1365.17**.**

Summary

1. We propose a systematic space-time code construction for joint channel estimation and error correction for a two-transmit-antenna and half-rate coding system.
2. Our systematic code construction, together with a fairly low-complexity ML decoder, then allows one to work with longer codes with no sacrifice in performance.
3. For codes of short block length, our simulations illustrate that the codes we propose have comparable performance to the best computer-searched codes.
4. For codes of long block lengths that are almost beyond the searchable range of existing computer systems, our codes are still better than some reference designs based on separate channel estimation and error correction components.

Conclusion and future works

Conclusion

- We propose systematical code designs for frequency-selective block fading channels, frequency-selective varying fading channels, frequency-non-selective varying fading channels, as well as space-time frequency-selective block fading channels, in which transmitter and receiver has no knowledge on CSI.
- When our codes are compared with the code designed by exhaustive computer searches in the literature, very limited performance difference can be observed.
- Unlike the structureless computer-searched codes, the structure of the code design we propose allows for ML priority-first search decoding, and hence, avoid exhaustive decoding.
- By simulations, the performances of our designed codes are shown comparable to the best codes found by computer search. Also, the decoding complexity of the proposed ML priority-first search decoding is significantly smaller when it is compared with the exhaustive checking decoding.

Future works

- In traditional packet-switched systems, frame synchronization is often achieved by the same training sequence.
- Without synchronization of the codeword margins, decoding may become technically infeasible.
- It is then fair to say that at this stage, the joint channel estimation and error correction codes may only fit well in an initial-sync, or circuit-switched, or TDD-based system environment.
- It will be an interesting, but quite challenging, future task to further enhance the proposed codes to possess self-synchronization capability.