Soft information aided parameter estimation

L. Vandendorpe

Thanks to C. Herzet, V. Ramon, A. Dejonghe, X. Wautelet, M. Moeneclaey (UGent)

UCL Communications and Remote Sensing Lab, Louvain-la-Neuve, Belgium

Outline

- Introduction/motivation
- The EM algorithm
- Coding and the MAP algorithm
- Synchronization of coded systems with the EM algorithm
- Illustration of performance
- CSI estimation for coded MIMO transmission
- Illustration and performance
- Cramer-Rao bound with coded/prior information

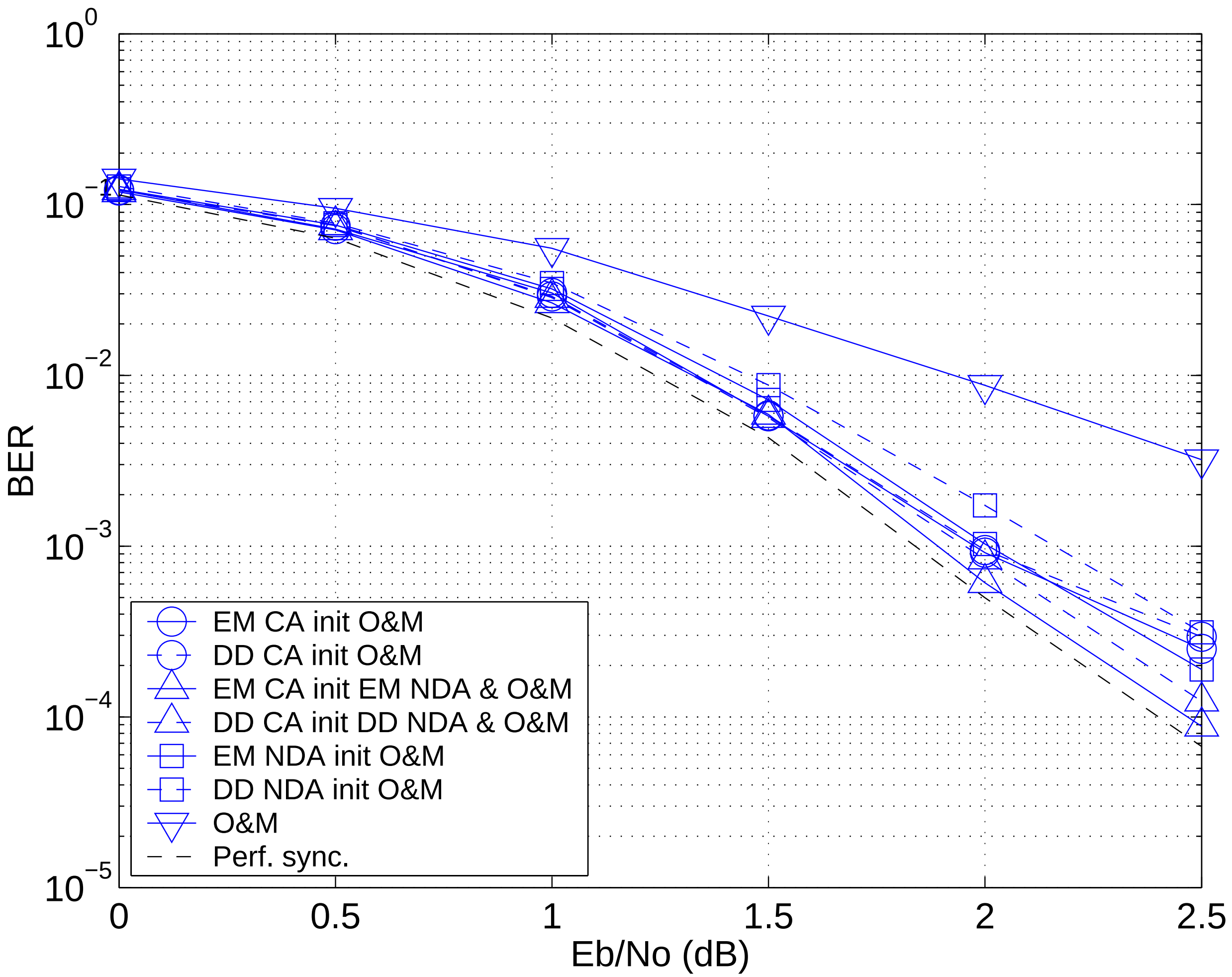
Outline

- **Introduction/motivation**
- The EM algorithm
- Coding and the MAP algorithm
- Synchronization of coded systems with the EM algorithm
- Illustration of performance
- CSI estimation for coded MIMO transmission
- Illustration and performance
- Cramer-Rao bound with coded/prior information

Motivation

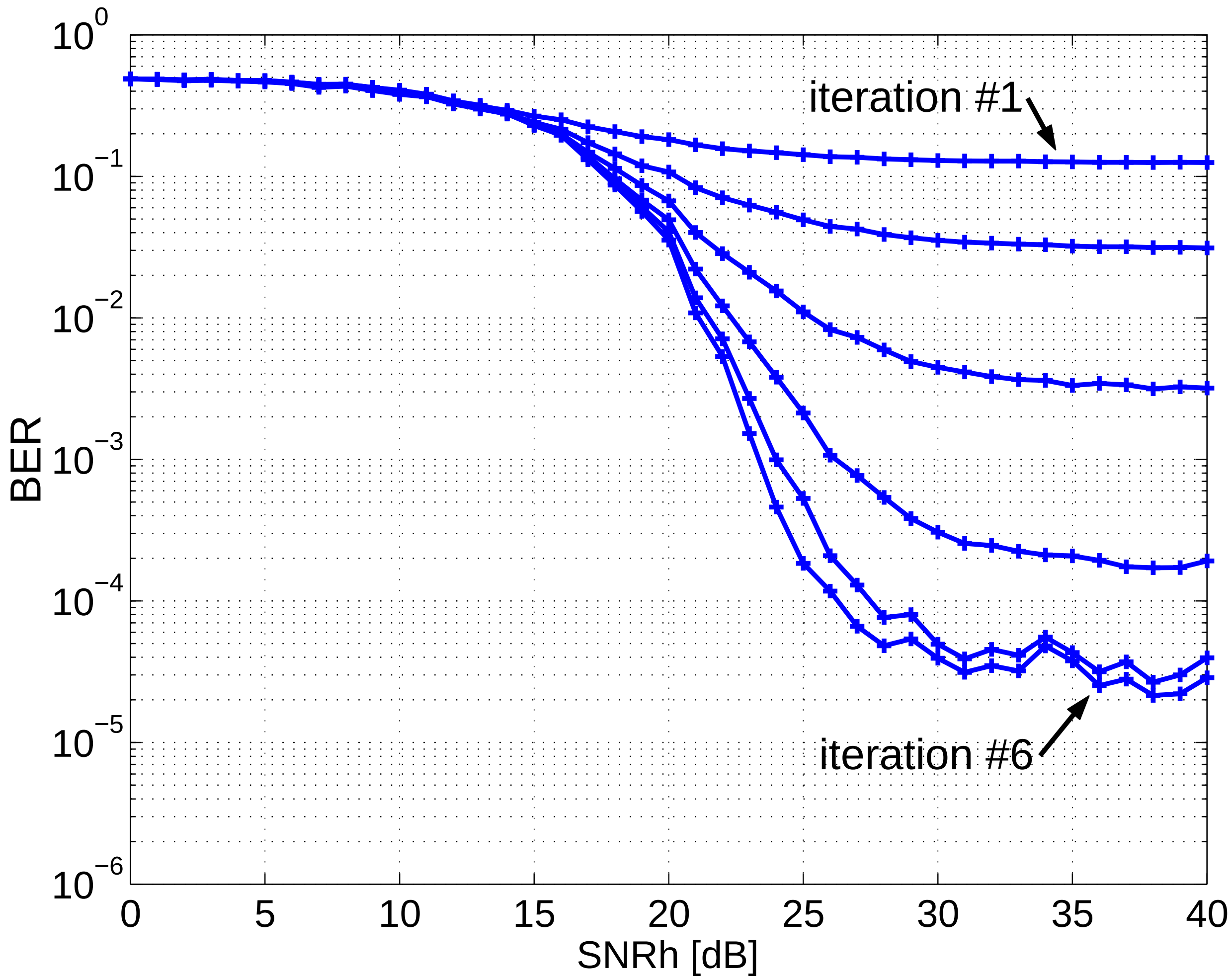
- Synchronization or parameter estimation required even if not primary goal (data)
- Synchronization/CSI required at the RX; CSI also of interest for TX
- Recent advances in coding (error correcting codes): operation point at (very) low SNRs; powerful with perfect sync.
- **Can we still reliably estimate parameters at low SNRs ?**
  - Increase of number of pilot symbols decreases spectral efficiency
  - Problem for short block transmission; use the information carried by the whole block
  - Turbo receivers (for instance) produce soft information
  - **How to use this soft information for sync/CSI estimation ?**
- The EM algorithm is a nice framework to derive soft-data aided estimation algorithms; adaptations are desirable however

Illustration: impact of timing estimation



- Turbo code performance for various timing synchronizers

Illustration: impact of CSI



- Turbo equalizer for BICM over Porat channel

Parameter estimation

- Assume data symbols a_k, observation vector $\mathbf{r}$, parameter vector θ
- Ultimate goal (min SER): detection/decoding given by

$$\begin{aligned} \hat{a}_k &= \arg\max_{\tilde{a}_k} \; p(\tilde{a}_k \,|\, \mathbf{r}) \\ &= \arg\max_{\tilde{a}_k} \; \int_\theta p(\tilde{a}_k \,|\, \mathbf{r}, \theta)\, p(\theta | \, \mathbf{r}) \mathrm{d}\theta \end{aligned} \tag{1}$$

- Suboptimal approach:

$$\hat{a}_k = \arg\max_{\tilde{a}_k} \; \int_\theta p(\tilde{a}_k \,|\, \mathbf{r}, \theta) \; p(\theta | \, \mathbf{r}) \mathrm{d}\theta \tag{2}$$

$$\simeq \arg\max_{\tilde{a}_k} \; p(\tilde{a}_k \,|\, \mathbf{r}, \theta = \arg\max_{\tilde{\theta}} \; p(\tilde{\theta} \,|\, \mathbf{r})) \tag{3}$$

Maximum likelihood parameter estimation

- Assume no prior information about parameters (uniform distribution)
- About the estimates:

$$\hat{\theta} = \arg\max_{\tilde{\theta}} \; p(\mathbf{r} \mid \tilde{\theta}) \tag{4}$$

$$= \arg\max_{\tilde{\theta}} \sum_{\mathbf{a}} p(\mathbf{r} \mid \mathbf{a}, \tilde{\theta}) \; p(\mathbf{a}) \tag{5}$$

- Function of the information we have about the transmitted sequence

ML parameter estimation: DA mode

- Assume one uses pilots only
- We transmit a sequence of pilot symbols $\mathbf{a}_{\text{pilot}}$

$$\hat{\theta} = \arg\max_{\tilde{\theta}} \; p(\mathbf{r}_{\text{pilot}} \,|\, \mathbf{a}_{\text{pilot}}, \tilde{\theta}) \tag{6}$$

- Easy to compute
- Only exploits part of the available information

ML parameter estimation: NDA mode

- All transmitted sequences assumed equiprobable

$$\hat{\theta} = \arg\max_{\tilde{\theta}} \sum_{\mathbf{a}} p(\mathbf{r} \mid \mathbf{a}, \tilde{\theta})\; p(\mathbf{a}) \quad (7)$$

$$= \arg\max_{\tilde{\theta}} \sum_{\mathbf{a}} p(\mathbf{r} \mid \mathbf{a}, \tilde{\theta})\; \left(\frac{1}{|\mathcal{A}|}\right)^{N} \quad (8)$$

$$(9)$$

- Untractable problem

ML parameter estimation: NDA mode

- All transmitted sequences assumed equiprobable

$$\hat{\theta} = \arg\max_{\tilde{\theta}} \sum_{\mathbf{a}} p(\mathbf{r} \,|\, \mathbf{a}, \tilde{\theta}) \; p(\mathbf{a}) \tag{10}$$

$$= \arg\max_{\tilde{\theta}} \sum_{\mathbf{a}} \underbrace{p(\mathbf{r} \,|\, \mathbf{a}, \tilde{\theta})}_{\text{low SNR approx.}} \; \left(\frac{1}{|\mathcal{A}|}\right)^{N} \tag{11}$$

$$\tag{12}$$

- Viterbi-Viterbi (phase), Oerder-Meyr (timing)

ML parameter estimation: Code aided mode

- Only existing codewords have non-zero probability:

$$\hat{\theta} = \arg\max_{\tilde{\theta}} \sum_{\mathbf{a}} p(\mathbf{r} \,|\, \mathbf{a}, \tilde{\theta})\ p(\mathbf{a}) \tag{13}$$

$$= \arg\max_{\tilde{\theta}} \sum_{\mathbf{a} \in \mathcal{B}} p(\mathbf{r} \,|\, \mathbf{a}, \tilde{\theta})\ p(\mathbf{a}) \tag{14}$$

- with $\mathcal{B} \subset \mathcal{A}^N$
- Untractable problem

Previous work (non exhaustive !)

- Basically two different paths are followed:
  - Parameter estimation can be embedded in the SISO module ("augmented trellis") [Colavolpe(2000)][Anastasopoulos,Chugg (2001)][Mielczarek(2002)]
  - Iterative detection/parameter estimation, coined turbo sync/parameter estimation
    - ∗ Carrier phase estimation in turbo coded systems: [Lottici, Luise (2002)]; [Burr (2002)]; [Oh,Cheun (2001)]; [Morlet (2000)]; [Langlais (2000)].
    - ∗ Timing recovery: [Mielczarek, Svensson (2002)]; [Li Zhang, Burr (2002)]
    - ∗ Channel estimation: [Kobayashi-Boutros-Caire (2001)], [Guenach2000], [Kaleh-Vallet (1994)]
  - The methods proposed for turbo-sync are rather "ad-hoc"
  - The EM framework provides a more structured approach

Outline

- Introduction/motivation
- **The EM algorithm**
- Coding and the MAP algorithm
- Synchronization of coded systems with the EM algorithm
- Illustration of performance
- CSI estimation for coded MIMO transmission
- Illustration and performance
- Cramer-Rao bound with coded/prior information

EM algorithm (1/3)

- Expectation-Maximization
- Seminal paper of [Dempster, Laird, Rubin, 1977]
- Can be used for the ML estimate or also the MAP estimate (Bayes framework, accounting for prior distribution)
- Example: assume observed data r and set of parameters to be estimated b
- The ML estimate of b is obtained as

$$\hat{b} = \arg\max_{b} p_r(r|b) \tag{15}$$

EM algorithm (2/3)

- Assume that instead of the *incomplete data* r one has access to the *complete data* z from which r may be obtained by a many-to-one mapping $r = H(z)$
- Definition of the complete data non unique; idea: $p_z(z|b)$ more easily obtained
- EM algorithms proceeds as follows
  - E-step (expectation): compute $\mathcal{Q}[b, \hat{b}^i] = \mathsf{E}[\ln p_z(z|b)|r, \hat{b}^i]$
  - M-step (maximization): solve $\hat{b}^{i+1} = \arg\max_b \mathcal{Q}[b, \hat{b}^i]$

EM algorithm (3/3)

- Idea: $\ln p_z(z|b)$ is not available; it is therefore a random variable and one maximizes its expectation given the observation r and the most recent value of the estimate $\hat{b}^i$
- Converges under mild conditions
- Can produce a local maximum
- Likelihood never decreases

Parameter estimation in the presence of nuisance (1/3)

- Let the complete data $\mathbf{r}$ denote a random vector obtained by expanding the received modulated-signal $r(t)$ onto a suitable basis and let $\mathbf{b}$ indicate a deterministic vector of parameters (sync parameters) to be estimated
- $\mathbf{r}$ also depends on a random discrete-valued nuisance parameter vector $\mathbf{a}$ independent of $\mathbf{b}$ and with a priori probability density function $p(\mathbf{a})$ (the data)
- Find the ML estimate $\hat{\mathbf{b}}$ of $\mathbf{b}$: $\hat{\mathbf{b}} = \arg\max_{\tilde{\mathbf{b}}}\{\ln p(\mathbf{r}|\tilde{\mathbf{b}})\}$, where

$$p(\mathbf{r}|\tilde{\mathbf{b}}) = \int_{\mathbf{a}} p(\mathbf{r}|\mathbf{a}, \tilde{\mathbf{b}})\, p(\mathbf{a})\, d\mathbf{a} \tag{16}$$

Parameter estimation in the presence of nuisance (2/3)

- Set $\mathbf{r}$ as the *incomplete* data set and $\mathbf{z} \triangleq [\mathbf{r}^T, \mathbf{a}^T]^T$ as the *complete* data set
- EM algorithm :

$$\mathcal{Q}(\tilde{\mathbf{b}}, \hat{\mathbf{b}}^{(n-1)}) = \int_{\mathbf{z}} p(\mathbf{z}|\mathbf{r}, \hat{\mathbf{b}}^{(\mathbf{n-1})}) \ \ln p(\mathbf{z}|\tilde{\mathbf{b}}) \ d\mathbf{z} \quad (17)$$

$$\hat{\mathbf{b}}^{(n)} = \arg\max_{\tilde{\mathbf{b}}} \{\mathcal{Q}(\tilde{\mathbf{b}}, \hat{\mathbf{b}}^{(n-1)})\} \quad (18)$$

Parameter estimation in the presence of nuisance (3/3)

- Using now the Bayes rule and taking into account the independence of $\mathbf{a}$ and $\mathbf{b}$ we may write

$$p(\mathbf{z}|\tilde{\mathbf{b}}) = p(\mathbf{r}, \mathbf{a}|\tilde{\mathbf{b}}) = p(\mathbf{r}|\mathbf{a}, \tilde{\mathbf{b}})\, p(\mathbf{a}|\tilde{\mathbf{b}}) = p(\mathbf{r}|\mathbf{a}, \tilde{\mathbf{b}})\, p(\mathbf{a}).$$

- It comes

$$\begin{aligned} \mathcal{Q}(\tilde{\mathbf{b}}, \hat{\mathbf{b}}^{(n-1)}) &= \int_{\mathbf{a}} p(\mathbf{a}|\mathbf{r}, \hat{\mathbf{b}}^{(n-1)}) \; \ln p(\mathbf{r}|\mathbf{a}, \tilde{\mathbf{b}}) \; d\mathbf{a} \\ &+ \int_{\mathbf{a}} p(\mathbf{a}|\mathbf{r}, \hat{\mathbf{b}}^{(n-1)}) \; \ln p(\mathbf{a}) \; d\mathbf{a}. \end{aligned} \tag{19}$$

- Finally, with the independence assumption

$$\mathcal{Q}(\tilde{\mathbf{b}}, \hat{\mathbf{b}}^{(n-1)}) = \int_{\mathbf{a}} p(\mathbf{a}|\mathbf{r}, \hat{\mathbf{b}}^{(\mathbf{n-1})}) \; \ln p(\mathbf{r}|\mathbf{a}, \tilde{\mathbf{b}}) \; d\mathbf{a}. \tag{20}$$

Parameter estimation in the presence of nuisance: comments

- Knowledge of a posteriori sequence (symbol) probabilities required

$$p(\mathbf{a}|\mathbf{r}, \hat{\mathbf{b}}^{(\mathbf{n-1})}) \tag{21}$$

- Should take into account the code information if any
- For convolutional code: can be computed exactly
- For turbo code or any iterative device, should be delivered after "a number" of iterations
- How do we get marginal a posteriori probabilities ?

Outline

- Introduction/motivation
- The EM algorithm
- **Coding and the MAP algorithm**
- Synchronization of coded systems with the EM algorithm
- Illustration of performance
- CSI estimation for coded MIMO transmission
- Illustration and performance
- Cramer-Rao bound with coded/prior information

How to improve coding (1/2) ?

- Classical codes:
  - ▷ block codes (BCH, Reed-Solomon,...)
  - ▷ convolutional codes (NSC, RSC)

  ⇒ Efficiency is increased by increasing the length of the codewords (block codes) or the code memory (convolutional codes).

  ⇒ Exponentially increasing complexity of the associated Maximum Likelihood (ML) decoding.

- Concatenated codes
  - ▷ Outer block code and inner convolutional code separated by an interleaver.
  - ▷ Separate decoding of the codes.

How to improve coding (2/2) ?

- Turbo-codes and iterative decoding (1995):

  ▷ Combination of several simple codes (constituent codes) in order to form a powerful global code.

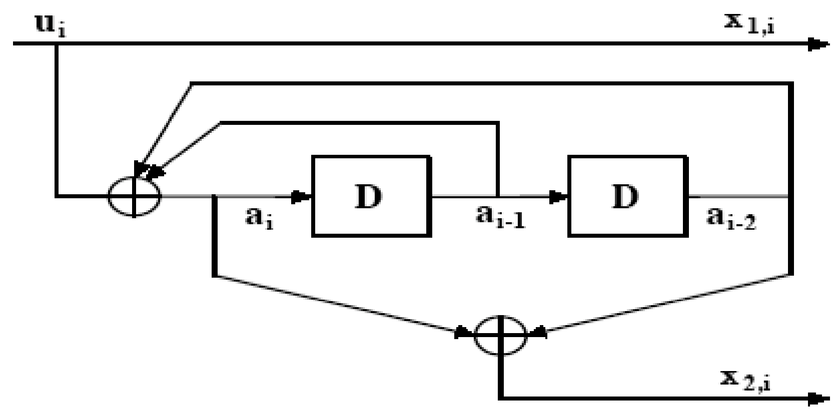
  ⇒ Attractive ML performances for the global code.

  ▷ Iterative decoding technique which allows the separate decoding of the constituent codes.

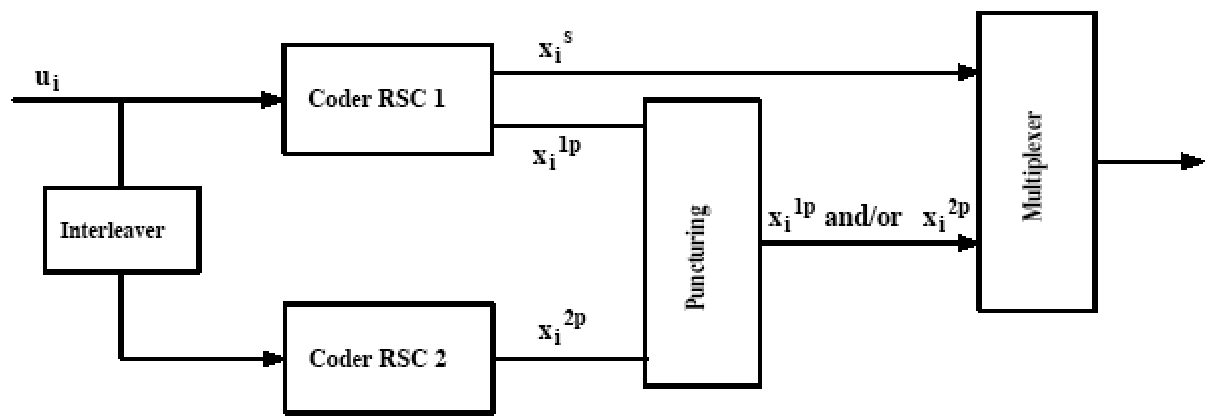
  ⇒ Performances close to those of the untractable ML decoding of the global code.

Classical turbo coding (1)

- rate-1/2 RSC code:



- Coding scheme:



Classical turbo coding (2)

- Parallel concatenation of 2 identical rate-1/2 RSC constituent codes.

- Pseudo-random interleaver: random permutation of the input sequence **u**.

  $\Rightarrow$ The two constituent encoders are coding the same information sequence **u** but in a different order.

- For each input binary information symbol u_i, we keep:

  $\triangleright$ the systematic output $x_i^s = u_i$ of the first RSC encoder.

  $\triangleright$ the coded outputs x_i^{1p} and x_i^{2p} of the two RSC encoders.

Classical turbo coding (3)

- The outputs are multiplexed to form the sequence:

$$\{\ldots, u_i, x_i^{1p}, x_i^{2p}, u_{i+1}, x_{i+1}^{1p}, x_{i+1}^{2p}, u_{i+2}, x_{i+2}^{1p}, x_{i+2}^{2p}, \ldots,\}$$

$\Rightarrow$ code rate $r = 1/3$.

- The code rate may be increased through puncturing.

$\Rightarrow$ Classically the code rate is increased to 1/2 as follows:

$$\{\ldots, u_i, x_i^{1p}, u_{i+1}, x_{i+1}^{2p}, u_{i+2}, x_{i+2}^{1p}, u_{i+3}, x_{i+3}^{2p}, \ldots,\}$$

- In practice, only the trellis of the first constituent code is terminated with negligible impact on the performances of the global turbo-code.

Decoding complexity

- Maximum Likelihood decoding of the global turbo-code ?

  ▷ $\mathcal{O}(2^N)$ complexity!

  N = information sequence length.

  ▷ Totally untractable!

⇒ Suboptimal iterative decoding technique (turbo-decoding).

  ▷ $\mathcal{O}(\mathrm{n}(2^K + 2^K))$ complexity!

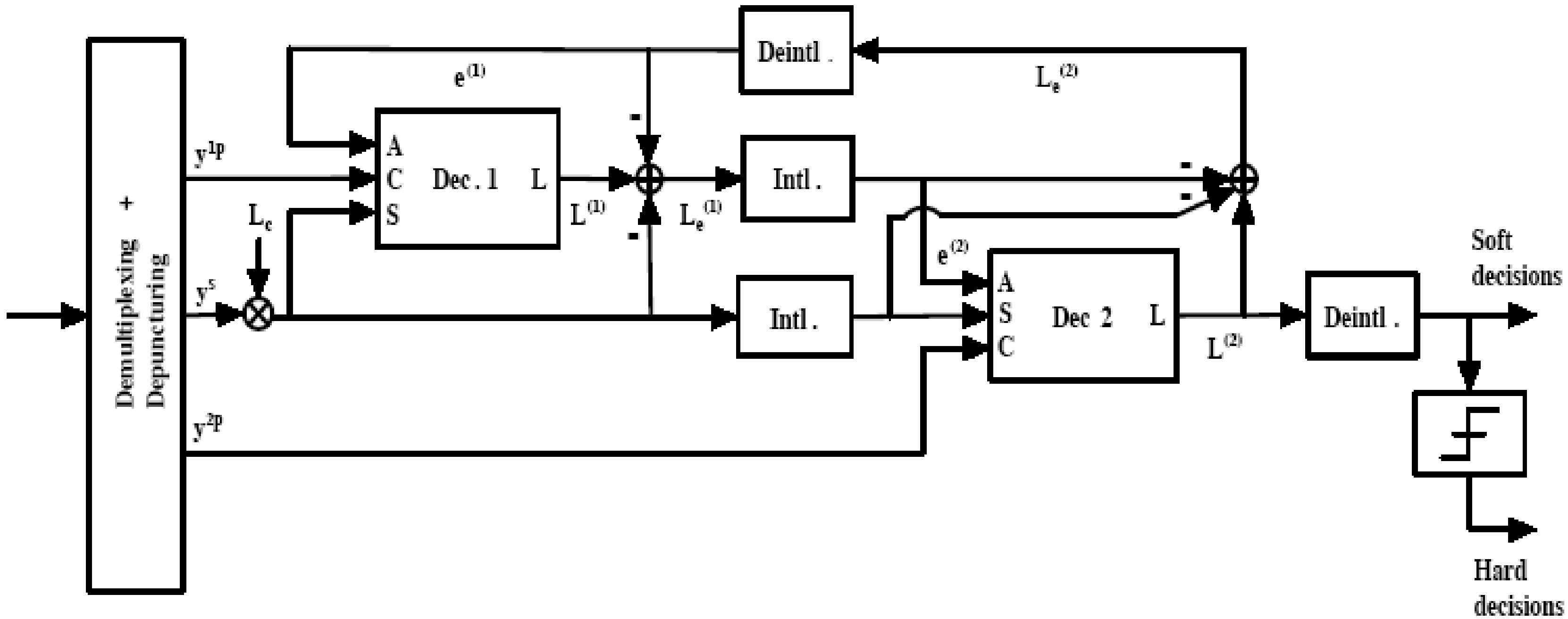
  K = constraint length of the constituent codes.

  n = number of iterations

  ▷ Performances (after convergence) close to those of ML decoding.

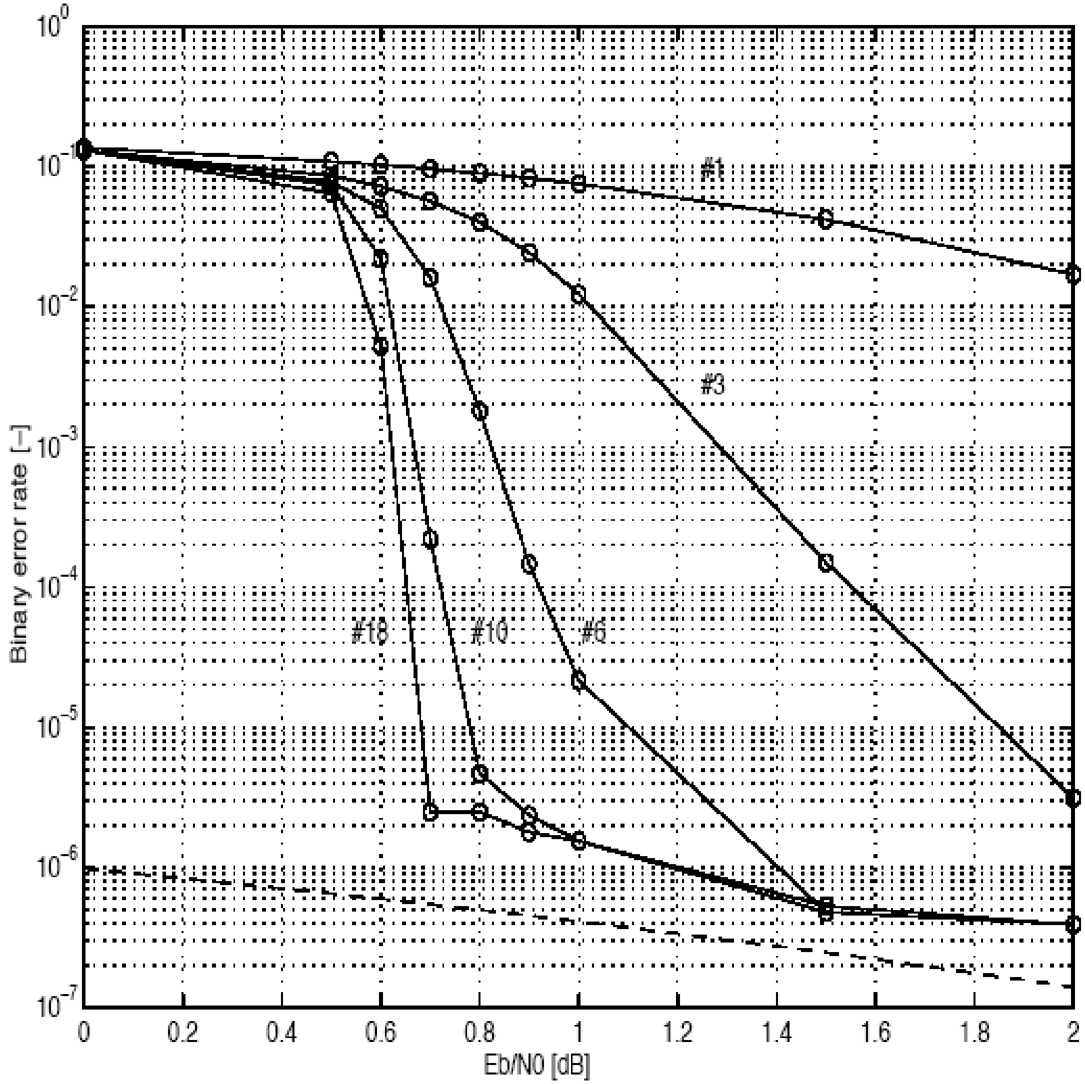
Iterative decoding

- Iterative decoding scheme:



- Soft information exchange between two soft-in/soft-out decoders.
- Progressive improvement in the reliability of the decisions.

Performance



Possible schemes

- Concatenation method:
  - ▷ Parallel concatenation of two or more constituent codes.
  - ▷ Serial concatenation of two or more constituent codes.
  - ▷ Hybrid concatenation of two or more constituent codes.

- Constituent codes:
  - ▷ rate-r convolutional codes (NSC or RSC).
  - ▷ rate-r block codes.

- In all cases:
  - ▷ Attractive asymptotic ML performances.
  - ▷ Iterative decoding.

Soft decisions
and
soft-in/soft-out (SISO)
decoding

Soft decisions (1)

- Hard decision:

  A discrete symbol from the input constellation is associated with each received sample at the demodulator.

- Soft decision:

  A continuous value is kept at the demodulator.

$\Rightarrow$ Reliability measure associated with the symbol.

$\Rightarrow$ Allows the full exploitation of the available information.

Soft decisions (2)

- Soft decision vs. hard decision: 2dB Gain!
- Soft decision in the binary case: *Log-Likelihood Ratio* (LLR).
- LLR of a discrete binary random variable $\mathbf{U}$:

$$L_U(u) = \ln\left(\frac{P_U(u=1)}{P_U(u=0)}\right)$$

Absolute value $\Rightarrow$ Reliability of the decision.

Sign $\Rightarrow$ hard decision.

$$\hat{u} = \begin{cases} 1 & \text{if } L_U(u) \geq 0 \\ 0 & \text{if } L_U(u) < 0 \end{cases}$$

Soft output of a channel (1)

- Information symbol $u \in \{0,1\}$ BPSK mapped to symbol $b \in \{+1,-1\}$.
- Memoryless channel associating the input symbol $b \in \{+1,-1\}$

  with the received sample y.

- The LLR of symbol u given the reception of symbol y is:

$$L(u|y) = \ln\left(\frac{P(u=1|y)}{P(u=0|y)}\right) = \ln\left(\frac{P(b=+1|y)}{P(b=-1|y)}\right)$$

Using the Bayes rule:

$$\begin{aligned} L(u|y) &= \ln\left(\frac{P(y|u=1)}{P(y|u=0)}\right) + \ln\left(\frac{P(u=1)}{P(u=0)}\right) \\ &= \ln\left(\frac{P(y|b=+1)}{P(y|b=-1)}\right) + \ln\left(\frac{P(u=1)}{P(u=0)}\right) \\ &= L_c y + L_a(u) \end{aligned}$$

Soft output of a channel (2)

- Two terms in $L(u|y)$:

  ▷ $L_c y$ is called *soft output of the channel.*

  Soft information associated with u, brought by the reception of y.

  ▷ $L_a(u)$ corresponds to the information available *a priori*

  at the receiver about u, independently of the reception of y.

- In the case of an AWGN channel, with noise variance σ^2:

$$\ln\left(\frac{P(y|b=+1)}{P(y|b=-1)}\right) = \ln\left(\frac{\exp(-\frac{1}{2\sigma^2}(b-1)^2}{\exp(-\frac{1}{2\sigma^2}(b+1)^2}\right) = \frac{2}{\sigma^2}y$$

$\Rightarrow$ The reliability value of the channel is given by $L_c = \frac{2}{\sigma^2}$.

SISO decoder (1)

- Decoder working with soft values at its inputs and outputs

$\Rightarrow$ *Soft-In/Soft-out* (SISO) decoder.

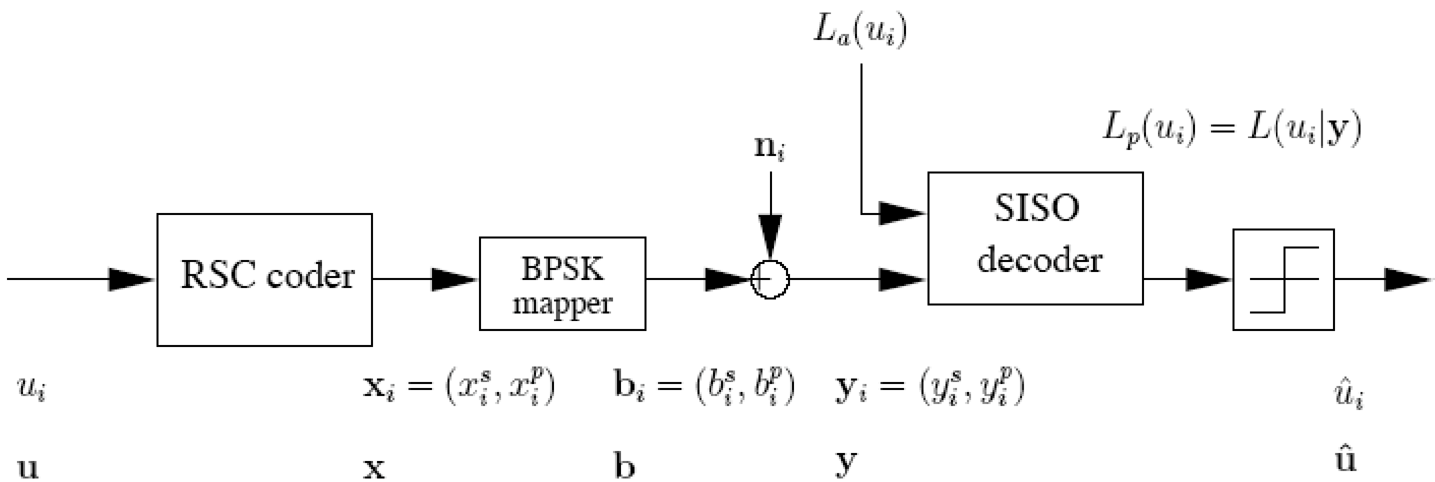
- Particular case here: rate-1/2 systematic code

  (straightforward generalization).

SISO decoder (2)

- Coder input: binary information symbols u_i $(i = 1, \ldots, N)$
- Coder output: coded symbols x_i^s, x_i^p.
- Coder output sequence: $\mathbf{x} = (\mathbf{x}_1, \ldots, \mathbf{x}_N)$ with $\mathbf{x}_i = (x_i^s, x_i^p)$.
- BPSK mapping $\Rightarrow$ sequence $\mathbf{b} = (\mathbf{b}_1, \ldots, \mathbf{b}_N)$

  with $\mathbf{b}_i = (b_i^s, b_i^p)$ and $b_i^s = 2x_i^s - 1$, $b_i^p = 2x_i^p - 1$.
- Channel $\Rightarrow$ output sequence $\mathbf{y} = (\mathbf{y}_1, \ldots, \mathbf{y}_N)$ with $\mathbf{y}_i = (y_i^s, y_i^p)$.



SISO decoder (3)

- Inputs of the SISO decoder:
  - ▷ Sequence $\mathbf{y}$ of the received symbols.

    Equivalently: sequences $\mathbf{y}^s = (y_1^s, \ldots, y_N^s)$ and $\mathbf{y}^p = (y_1^p, \ldots, y_N^p)$.

    Equivalently: sequences of soft channel values $L_c\mathbf{y}^s$ and $L_c\mathbf{y}^p$.
  - ▷ Sequence $\mathbf{L}_a$ of a priori information about

    the information symbols $\{u_i\}$ $(i = 1, \ldots, N)$:

$$L_a(u_i) = \ln\left(\frac{P(u_i = 1)}{P(u_i = 0)}\right)$$

- Output of the SISO decoder:
  - ▷ LLR of the a posteriori probabilities of the information symbols:

$$L_p(u_i) = \ln\left(\frac{P(u_i = 1|\mathbf{y})}{P(u_i = 0|\mathbf{y})}\right) = \ln\left(\frac{P(u_i = 1|\mathbf{y}^s, \mathbf{y}^p)}{P(u_i = 0|\mathbf{y}^s, \mathbf{y}^p)}\right)$$

SISO decoder (4)

- A SISO decoder is implemented with algorithms able to estimate the symbol a posteriori probabilities.

- From SISO decoder output, decoded symbols obtained via hard decision:

$$\hat{u}_i = \begin{cases} 1 & \text{if } L_p(u_i) \geq 0 \\ 0 & \text{if } L_p(u_i) < 0 \end{cases}$$

- SISO decoder + hard decision ⇒ symbol-by-symbol MAP decoding:

$$\hat{u}_i = \arg\max_u P(u|\mathbf{y})$$

- Fundamental property (SYSTEMATIC CODE):

$$L_p(u_i) = (L_c y_i^s) + L_a(u_i) + L_e(u_i)$$

⇒ The *a posteriori* LLR $L_p(u_i)$ can be split into three terms.

SISO decoder (5)

⇒ The *a posteriori* LLR $L_p(u_i)$ can be split into three terms:

▷ $L_c y_i^s$: information about symbol $x_i^s = u_i$ through direct (noisy) observation at the output of the channel.

▷ $L_a(u_i)$: a priori information about the information symbol u_i.

▷ $L_e(u_i)$: *extrinsic information* about the information symbol u_i.

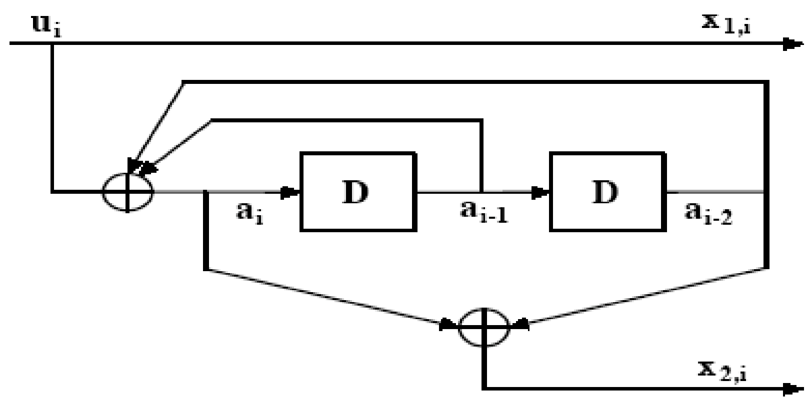
⇒ Supply of soft information brought by the decoding process.

⇒ Depends on y_m^s $(m = 1, \ldots, N; m \neq i)$, y_m^p $(m = 1, \ldots, N)$, $L_a(u_m)$ $(m = 1, \ldots, N; m \neq i)$.

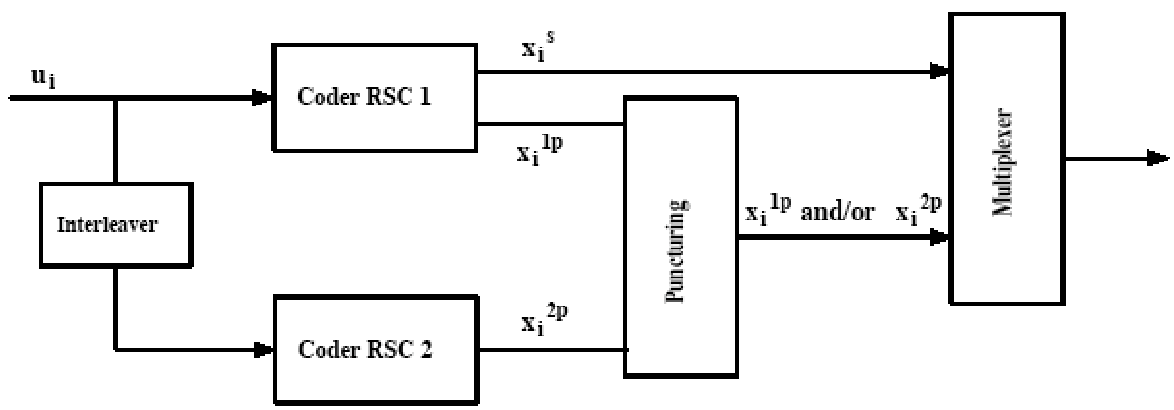
Iterative decoding

Classical turbo coding scheme

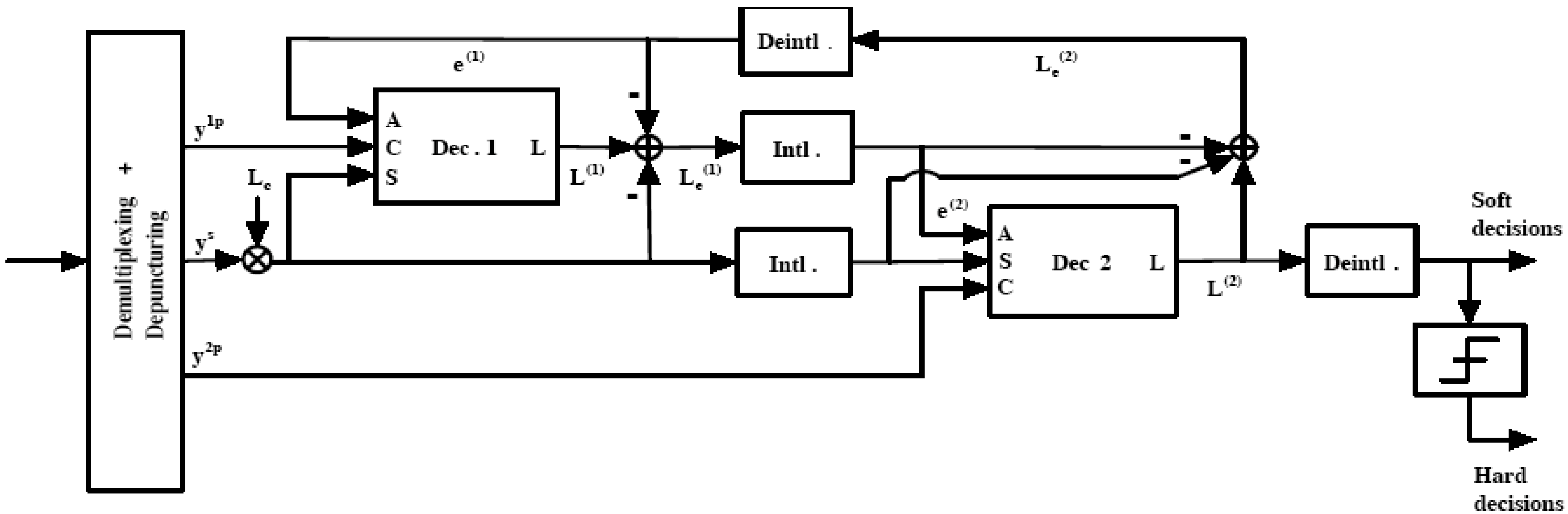
- rate-1/2 RSC code:



- Coding scheme:



Iterative decoding (1)



- Demultiplexing $\Rightarrow$ sequence $\mathbf{y}^s$ (systematic output of CC1), sequences $\mathbf{y}^{1p}$ and $\mathbf{y}^{2p}$ (coded outputs of CC1 and CC2).
- If puncturing: missing values are replaced by 0.

Iterative decoding (2)

- Decoding scheme based on the association of 2 SISO decoders corresponding to the 2 constituent codes of the turbo-code.
- These SISO decoders collaborate through an extrinsic information exchange.
- Iterative processing leads to progressive increase in the reliability of the decisions.
- Performances close (after convergence) to those of the untractable ML decoding of the turbo-code.

Iterative decoding (3)

- The first decoder ensures the decoding of the first constituent code based on the received sequences $\mathbf{y}^s$, $\mathbf{y}^{1p}$ and on the a priori information sequence $\mathbf{L}_a^{(1)}$ about the transmitted symbols.

- At the first iteration: no a priori information $\Rightarrow L_a^{(1)}(u_i) = 0 \quad \forall i$.

- It outputs a sequence $\mathbf{L}_p^{(1)}$ of a posteriori LLRs $L_p^{(1)}(u_i)$:

$$L_p^{(1)}(u_i) = \ln\left(\frac{P(u_i = 1|\mathbf{y}^s, \mathbf{y}^{1p})}{P(u_i = 0|\mathbf{y}^s, \mathbf{y}^{1p})}\right)$$

- The extrinsic component $\mathbf{L}_e^{(1)}$ is then extracted from the output $\mathbf{L}_p^{(1)}$:

$$L_e^{(1)}(u_i) = L_p^{(1)}(u_i) - L_c y_i^s - L_a^{(1)}(u_i)$$

Iterative decoding (4)

- The second decoder ensures the decoding of the second constituent code based on the received sequences $\mathbf{y}^s$ (interleaved), $\mathbf{y}^{2p}$ and on the a priori information sequence $\mathbf{L}_a^{(2)}$ about the transmitted symbols.

- $\mathbf{L}_a^{(2)}$ is obtained by interleaving of the extrinsic information sequence $\mathbf{L}_e^{(1)}$ produced by decoder 1.

- The second decoder outputs a sequence $\mathbf{L}_p^{(2)}$ of a posteriori LLRs $L_p^{(2)}(u_j)$:

$$L_p^{(2)}(u_j) = \ln\left(\frac{P(u_j = 1|\mathbf{y}^s, \mathbf{y}^{2p})}{P(u_j = 0|\mathbf{y}^s, \mathbf{y}^{2p})}\right)$$

- Again, the extrinsic component $\mathbf{L}_e^{(2)}$ is extracted from the output $\mathbf{L}_p^{(2)}$:

$$L_e^{(2)}(u_j) = L_p^{(2)}(u_j) - L_c y_j^s - L_a^{(2)}(u_j)$$

Iterative decoding (5)

- A second iteration may now begin:

  The sequence $\mathbf{L}_e^{(2)}$ of extrinsic information produced by decoder 2 becomes (after deinterleaving) the sequence $\mathbf{L}_a^{(1)}$ of a priori information for the decoder 1.

- The fundamental principle is that the extrinsic information provided by one of the decoders becomes the a priori information for the other.

$\Rightarrow$ Improved quality of the decoding for each of the SISO decoders.

- Through iterations: progressive increase in the reliability of the decisions.

Iterative decoding (6)

- At the last iteration, the best estimation available about the transmitted symbols is given by the deinterleaved a posteriori output of the second decoder.

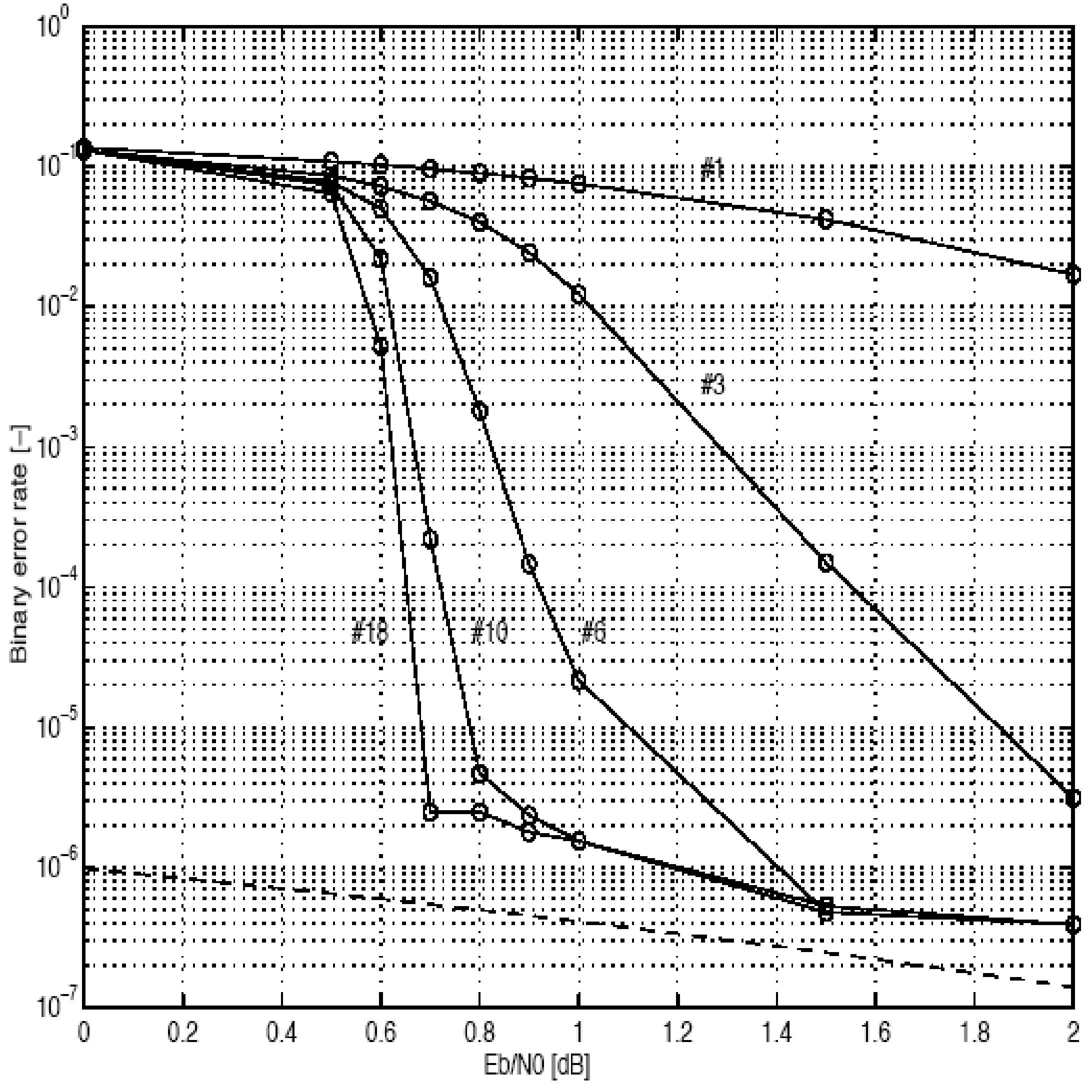
- The final hard decision is:

$$\hat{u}_i = \begin{cases} 1 & \text{if } L_p^{(2)}(u_i) \geq 0 \\ 0 & \text{if } L_p^{(2)}(u_i) < 0 \end{cases}$$

Iterative decoding (7)

- This scheme will perform efficiently if the two SISO decoders are decorrelated information sources one for each other.
- This decorrelation is possible thanks to the interleaver.
- This is also the reason why only the extrinsic part of the a posteriori LLRs at the output of the SISO decoders is used during the exchange process.

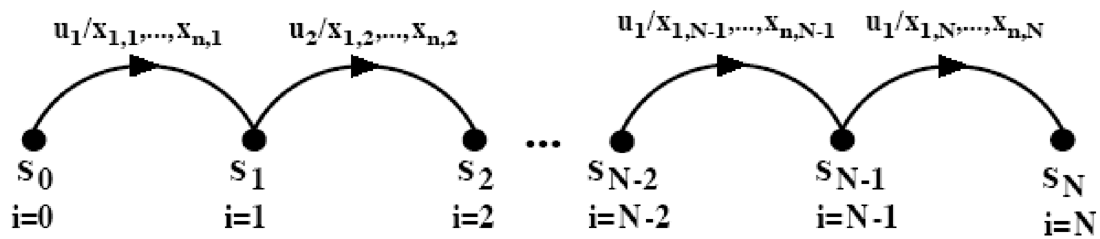
Performance



Symbol by symbol algorithm

Markov process

- Markov process:
  - ▷ State s_i in finite set $\mathcal{S}$ a each time i $(i = 0, \ldots, N)$.
  - ▷ Input: sequence $\mathbf{u}$, output: sequence $\mathbf{x}$.
  - ▷ Particluar case: 1 input symbol, n output symbols:



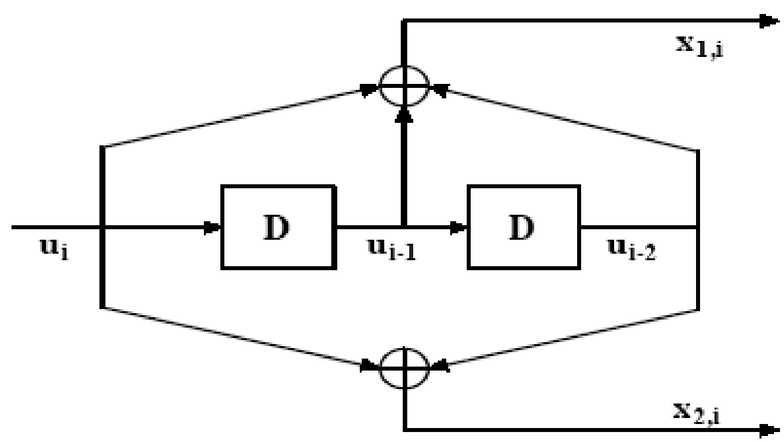
  - At time i, transition between states $s_{i-1} = s'$ and $s_i = s$ caused by symbol u_i $(i = 1, \ldots, N)$ generates symbols $\mathbf{x}_i = (x_{i,1}, \ldots, x_{i,n})$ of sequence $\mathbf{x}$.

- Fundamental property:

$$P(s_i | s_{i-1}, \ldots, s_0) = P(s_i | s_{i-1})$$

Convolutional code (1)

- Convolutional code = Markov process



- State = content of the shift-registers.

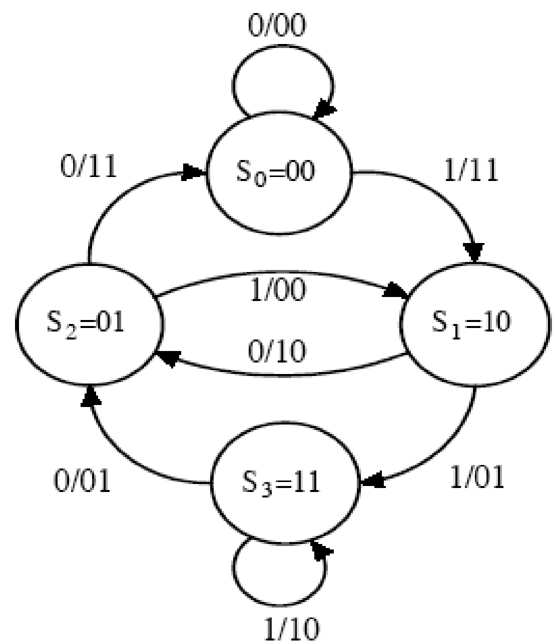
$\Rightarrow$ In the case of an NSC code:

$$s_i = (u_i, \ldots, u_{i-M+1})$$

- Memory $M \Rightarrow 2^M$ possible states S_j $(j = 0, \ldots, 2^M - 1)$.

Convolutional code (2)

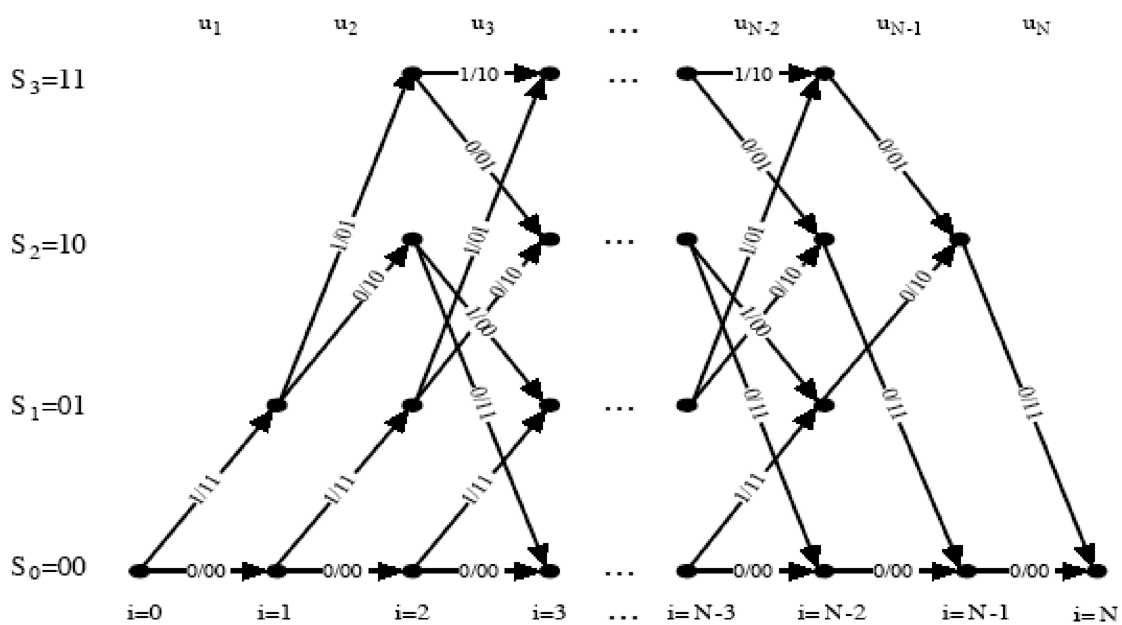
- State diagram representation of a convolutional code:



- Encoding of a sequence ⇒ path through the state diagram.

Convolutional code (3)

- Trellis representation of a convolutional code:



- Encoding of a sequence ⇒ path through the trellis diagram.

Transmission scheme (1)

- Rate $r = 1/n$ convolutional encoder.
- Memory M encoder $\Rightarrow 2^M$ possible states in set $\mathcal{S}$.
- Coder state at timestep i: s_i.
- At timestep i, transition (s', s) between states $s_{i-1} = s'$ and $s_i = s$.

- Input: binary information symbols u_i $(i = 1, \ldots, N)$
- Output: coded symbols $x_{i,1}, \ldots, x_{i,n}$.
- Output sequence: $\mathbf{x} = (\mathbf{x}_1, \ldots, \mathbf{x}_N)$ with $\mathbf{x}_i = (x_{i,1}, \ldots, x_{i,n})$.

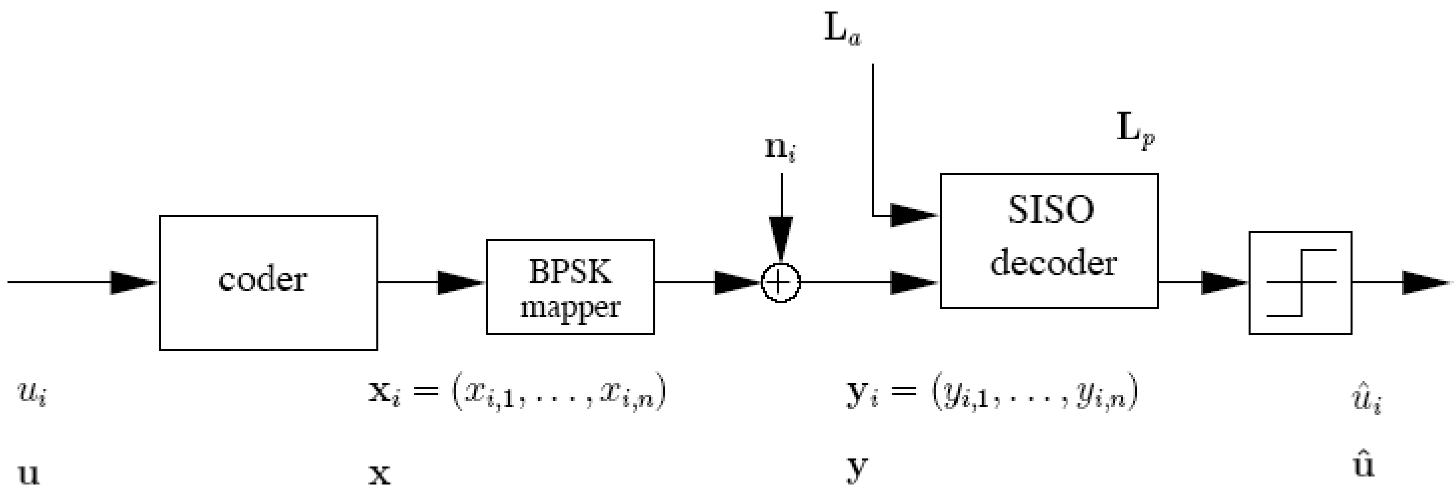
- BPSK mapping $\Rightarrow$ sequence $\mathbf{b} = (\mathbf{b}_1, \ldots, \mathbf{b}_N)$

  with $\mathbf{b}_i = (b_{i,1}, \ldots, b_{i,n})$ and $b_{i,j} = 2x_{i,j} - 1$.

- Channel $\Rightarrow$ output sequence $\mathbf{y} = (\mathbf{y}_1, \ldots, \mathbf{y}_N)$ with $\mathbf{y}_i = (y_{i,1}, \ldots, y_{i,n})$.

Transmission scheme (2)

- Transmission scheme:



SISO decoder

- Input of the SISO decoder:
  - ▷ Received sequence $\mathbf{y}$.
  - ▷ A priori LLR sequence $\mathbf{L}_a$ with entries $L_a(u_i) = \ln \frac{P(u_i=1)}{P(u_i=0)}$.
- Output of the SISO decoder:
  - ▷ A posteriori LLR sequence $\mathbf{L}_p$ with entries $L_p(u_i) = \ln \frac{P(u_i=1|\mathbf{y})}{P(u_i=0|\mathbf{y})}$.
- Data:
  - ▷ initial state s_0 and final state s_N.
  - ▷ Code trellis.
  - ▷ Noise variance σ^2.

BCJR algorithm (1)

- Symbol-by-symbol a posteriori probability (APP) evaluation

$\Leftrightarrow$ Minimization of the symbol error rate $\Rightarrow$ optimal!

- BCJR algorithm (1974):

  Evaluation of the a posteriori probabilities of the states and transitions of a Markov source observed through a discrete-time memoryless channel.

BCJR algorithm (2)

- The BCJR algorithm provides the a posteriori states and transitions probabilities:

$$P(s_i = s|\mathbf{y}) \text{ or } P(s_i = s, \mathbf{y})$$

and:

$$P(s_{i-1} = s', s_i = s|\mathbf{y}) \text{ or } P(s_{i-1} = s', s_i = s, \mathbf{y})$$

on the basis of:

$\Rightarrow$ the received sequence: $\mathbf{y}$.

$\Rightarrow$ the channel type $\rightarrow p(\mathbf{y}_i|s_{i-1} = s', s_i = s)$.

$\Rightarrow$ the transitions a priori probabilities: $p(s_i = s|s_{i-1} = s')$.

- Slight modification necessary to obtain a SISO decoder.

$\Rightarrow$ "MAP" algorithm.

MAP algorithm (1)

- Slight modification of the BCJR algorithm ⇒ "MAP" algorithm.

- The goal of the MAP algorithm is to provide an APP LLR (soft output):

$$L_p(u_i) = \ln\left(\frac{P(u_i = 1|\mathbf{y})}{P(u_i = 0|\mathbf{y})}\right)$$

  based on the received sequence $\mathbf{y}$ and the a priori information sequence $\mathbf{L}_a$.

⇒ Optimal algorithm for the implementation of a SISO decoder.

- Combined with hard detection, it realizes MAP decoding:

$$\hat{u}_i = \begin{cases} 1 & \text{if } L_p(u_i) \geq 0 \\ 0 & \text{if } L_p(u_i) < 0 \end{cases}$$

  Equivalent to:

$$\hat{u}_i = \arg\max_u P(u|\mathbf{y})$$

MAP algorithm (2)

- The MAP algorithm provides the a posteriori LLR:

$$L_p(u_i) = \ln\left(\frac{P(u_i = 1|\mathbf{y})}{P(u_i = 0|\mathbf{y})}\right)$$

- As the knowledge of $s_{i-1} = s'$ and $s_i = s$ determines u_i, we have

$$L_p(u_i) = \ln\left(\frac{\sum_{\mathcal{S}^+} p(s_{i-1} = s', s_i = s|\mathbf{y})}{\sum_{\mathcal{S}^-} p(s_{i-1} = s', s_i = s|\mathbf{y})}\right)$$

  where S_+ (resp. S_-) is the set of transitions $(s_{i-1} = s', s_i = s)$ caused by a symbol $u_i = 1$ (resp. $u_i = 0$).

- This can be simplified as:

$$L_p(u_i) = \ln\left(\frac{\sum_{\mathcal{S}^+} p(s_{i-1} = s', s_i = s, \mathbf{y})}{\sum_{\mathcal{S}^-} p(s_{i-1} = s', s_i = s, \mathbf{y})}\right)$$

MAP algorithm (3)

- The probability $p(s_{i-1} = s', s_i = s, \mathbf{y})$ is computed as (BCJR algorithm):

$$
\begin{aligned}
& p(s_{i-1} = s', s_i = s, \mathbf{y}) \\
&= p(s_{i-1} = s', \mathbf{y}_{j<i}) p(\mathbf{y}_{j\geq i}, s_i = s | s_{i-1} = s', \mathbf{y}_{j<i}) \\
&= p(s_{i-1} = s', \mathbf{y}_{j<i}) p(\mathbf{y}_{j\geq i}, s_i = s | s_{i-1} = s') \\
&= p(s_{i-1} = s', \mathbf{y}_{j<i}) p(\mathbf{y}_i, \mathbf{y}_{j>i}, s_i = s | s_{i-1} = s') \\
&= p(s_{i-1} = s', \mathbf{y}_{j<i}) \frac{p(\mathbf{y}_i, \mathbf{y}_{j>i}, s_i = s, s_{i-1} = s')}{p(s_{i-1} = s')} \\
&= p(s_{i-1} = s', \mathbf{y}_{j<i}) \frac{p(\mathbf{y}_i, s_i = s, s_{i-1} = s')}{p(s_{i-1} = s')} p(\mathbf{y}_{j>i} | \mathbf{y}_i, s_i = s, s_{i-1} = s') \\
&= p(s_{i-1} = s', \mathbf{y}_{j<i}) p(\mathbf{y}_i, s_i = s | s_{i-1} = s') p(\mathbf{y}_{j>i} | s_i = s) \\
&= p(s_{i-1} = s', \mathbf{y}_{j<i}) p(\mathbf{y}_i | s_{i-1} = s', s_i = s) P(s_i = s | s_{i-1} = s') p(\mathbf{y}_{j>i} | s_i = s)
\end{aligned}
$$

NB: if $s_i = s$ is known, events after time i do not depend on $\mathbf{y}_{j<i+1}$.

MAP algorithm (4)

- Defining:

  ▷ $\alpha_{i-1}(s') = p(s_{i-1} = s', \mathbf{y_{j<i}})$,

  ▷ $\beta_i(s) = p(\mathbf{y_{j>i}} | s_i = s)$,

  ▷ $\gamma_i(s', s) = p(\mathbf{y}_i, s_i = s | s_{i-1} = s')$

  $= p(\mathbf{y_i} | s_{i-1} = s', s_i = s) p(s_i = s | s_{i-1} = s')$,

We have:

$$p(s_{i-1} = s', s_i = s, \mathbf{y}) = \alpha_{i-1}(s').\gamma_i(s', s).\beta_i(s)$$

MAP algorithm (5)

- Parameters α are computed as follows:

$$
\begin{aligned}
\alpha_i(s) &= p(s_i = s, \mathbf{y}_{j<i+1}) \\
&= \sum_{s' \in \mathcal{S}} p(s_{i-1} = s', s_i = s, \mathbf{y}_{j<i+1}) \\
&= \sum_{s' \in \mathcal{S}} p(s_{i-1} = s', s_i = s, \mathbf{y}_{j<i}, \mathbf{y}_i) \\
&= \sum_{s' \in \mathcal{S}} p(s_{i-1} = s', \mathbf{y}_{j<i}) p(s_i = s, \mathbf{y}_i | s_{i-1} = s', \mathbf{y}_{j<i}) \\
&= \sum_{s' \in \mathcal{S}} p(s_{i-1} = s', \mathbf{y}_{j<i}) p(s_i = s, \mathbf{y}_i | s_{i-1} = s') \\
&= \sum_{s' \in \mathcal{S}} \alpha_{i-1}(s').\gamma_i(s', s)
\end{aligned}
$$

MAP algorithm (6)

- Parameters α are obtained via a forward recursion:

$$\alpha_i(s) = \sum_{s' \in \mathcal{S}} \alpha_{i-1}(s').\gamma_i(s', s)$$

for $(i = 0, \ldots, N-1)$ and $\forall s \in \mathcal{S}$.

- The initial conditions are:

$$\alpha_0(s_0) = 1 \text{ and } \alpha_0(s \neq s_0) = 0$$

$\Leftrightarrow$ The initial state is known to be s_0.

MAP algorithm (7)

- Parameters β are computed as follows:

$$
\begin{aligned}
\beta_{i-1}(s') &= p(\mathbf{y}_{j>i-1}|s_{i-1}=s') \\
&= \sum_{s\in\mathcal{S}} p(s_i=s,\mathbf{y}_{j>i-1}|s_{i-1}=s') \\
&= \sum_{s\in\mathcal{S}} p(s_i=s,\mathbf{y}_{j>i},\mathbf{y}_i|s_{i-1}=s') \\
&= \sum_{s\in\mathcal{S}} \frac{p(s_i=s,\mathbf{y}_{j>i},\mathbf{y}_i,s_{i-1}=s')}{p(s_{i-1}=s')} \\
&= \sum_{s\in\mathcal{S}} p(\mathbf{y}_{j>i}|s_i=s,\mathbf{y}_i,s_{i-1}=s')\frac{p(s_i=s,\mathbf{y}_i,s_{i-1}=s')}{p(s_{i-1}=s')} \\
&= \sum_{s\in\mathcal{S}} p(\mathbf{y}_{j>i}|s_i=s)p(s_i=s,\mathbf{y}_i|s_{i-1}=s') \\
&= \sum_{s\in\mathcal{S}} \beta_i(s).\gamma_i(s',s)
\end{aligned}
$$

MAP algorithm (8)

- Parameters β are obtained via a backward recursion:

$$\beta_{i-1}(s') = \sum_{s \in \mathcal{S}} \beta_i(s).\gamma_i(s', s)$$

for $(i = 2, \ldots, N+1)$ and $\forall s' \in \mathcal{S}$.

- If trellis termination, the initial conditions are:

$$\beta_N(s_N) = 1 \text{ and } \beta_N(s \neq s_N) = 0$$

$\Leftrightarrow$ The final state is known to be s_N.

- If no trellis termination, the initial conditions are:

$$\beta_N(s) = \frac{1}{\#\mathcal{S}} \; \forall s \in \mathcal{S}$$

$\Leftrightarrow$ The final state is unknown.

MAP algorithm (9)

- $\gamma_i(s', s)$ associated with a transition between states $s_{i-1} = s'$ and $s_i = s$:

$$\begin{aligned}\gamma_i(s', s) &= p(\mathbf{y}_i, s_i = s | s_{i-1} = s') \\ &= p(\mathbf{y_i} | s_{i-1} = s', s_i = s).P(s_i = s | s_{i-1} = s')\end{aligned}$$

  In terms of symbols:

$$\gamma_i(s', s) = p(\mathbf{y_i} | u_i, s_{i-1} = s').P(u_i)$$

▷ $p(\mathbf{y_i} | u_i, s_{i-1} = s')$ is evaluated on the basis of the received symbol and the channel type.

▷ $P(u_i)$ is evaluated on the basis of the a priori information $L_a(u_i)$.

- $\gamma_i(s', s)$ = metric associated with the transition $(s_{i-1} = s', s_i = s)$.

  The same as in MAP sequence estimation and SOVA.

MAP algorithm: summary

- The MAP algorithm computes the a posteriori LLR $L_p(u_i)$

  of the information bits u_i (for $i = 1, \ldots, N$):

$$L_p(u_i) = \ln\left(\frac{\sum_{\mathcal{S}^+} \alpha_{i-1}(s').\gamma_i(s',s).\beta_i(s)}{\sum_{\mathcal{S}^-} \alpha_{i-1}(s').\gamma_i(s',s).\beta_i(s)}\right) \qquad (i = 1, \ldots, N)$$

- $\alpha \Rightarrow$ forward recursion with appropriate initial condition:

$$\alpha_i(s) = \sum_{s' \in \mathcal{S}} \alpha_{i-1}(s').\gamma_i(s',s) \qquad (i = 0, \ldots, N-1; \forall s \in \mathcal{S})$$

- $\beta \Rightarrow$ backward recursion with appropriate initial condition:

$$\beta_{i-1}(s') = \sum_{s \in \mathcal{S}} \beta_i(s).\gamma_i(s',s) \qquad (i = 2, \ldots, N+1; \forall s' \in \mathcal{S})$$

- $\gamma \Rightarrow$ calculated based on the received symbols and the a priori information:

$$\overline{\gamma}_i(s',s) = p(\mathbf{y_i}|s_{i-1} = s', s_i = s).P(s_i = s|s_{i-1} = s') \qquad \forall i; \forall (s',s) \in \text{trellis}$$

MAP algorithm: log MAP

- The MAP algorithm has numerical problems.

$\Rightarrow$ Implementation in the logarithmic domain:

- Define $\overline{\alpha}_i(s) = \ln(\alpha_i(s))$, $\overline{\beta}_i(s) = \ln(\beta_i(s))$ and $\overline{\gamma}_i(s', s) = \ln(\gamma_i(s', s))$.

- The a posteriori LLR becomes:

$$\begin{aligned}
L_p(u_i) &= \ln\left(\frac{\sum_{\mathcal{S}^+} \exp\left(\overline{\alpha}_{i-1}(s')\right) . \exp\left(\overline{\gamma}_i(s', s)\right) . \exp(\overline{\beta}_i(s))}{\sum_{\mathcal{S}^-} \exp\left(\overline{\alpha}_{i-1}(s')\right) . \exp\left(\overline{\gamma}_i(s', s)\right) . \exp(\overline{\beta}_i(s))}\right) \\
&= \ln\left(\sum_{\mathcal{S}^+} \exp\left(\overline{\alpha}_{i-1}(s') + \overline{\gamma}_i(s', s) + \overline{\beta}_i(s)\right)\right) \\
&- \ln\left(\sum_{\mathcal{S}^-} \exp\left(\overline{\alpha}_{i-1}(s') + \overline{\gamma}_i(s', s) + \overline{\beta}_i(s)\right)\right)
\end{aligned}$$

MAP algorithm: max log MAP

- Using the approximation:

$$\ln\left(\exp(x) + \exp(y) + \exp(z)\right) \approx \max(x, y, z)$$

we have

$$\begin{aligned} L_p(u_i) &\approx \max_{\mathcal{S}^+}(\overline{\alpha}_{i-1}(s') + \overline{\gamma}_i(s', s) + \overline{\beta}_i(s)) \\ &- \max_{\mathcal{S}^-}(\overline{\alpha}_{i-1}(s') + \overline{\gamma}_i(s', s) + \overline{\beta}_i(s)) \end{aligned}$$

▷ The forward recursion for parameters $\overline{\alpha}_i(s)$ becomes:

$$\overline{\alpha}_i(s) = \max_{s' \in \mathcal{S}}(\overline{\alpha}_{i-1}(s') + \overline{\gamma}_i(s', s))$$

with initial conditions :

$$\overline{\alpha}_0(s_0) = 0 \text{ and } \overline{\alpha}_0(s \neq s_0) = -\infty$$

MAP algorithm: max log MAP

▷ The backward recursion for parameters $\overline{\beta}_{i-1}(s')$ becomes:

$$\overline{\beta}_{i-1}(s) = \max_{s \in \mathcal{S}}(\overline{\beta}_i(s) + \overline{\gamma}_i(s', s))$$

with initial conditions:

$$\overline{\beta}_N(s_N) = 0 \text{ and } \overline{\beta}_N(s \neq s_N) = -\infty \quad \text{if trellis termination}$$

or initial conditions:

$$\overline{\beta}_N(s) = \ln(\frac{1}{\#\mathcal{S}})\ \forall s \in \mathcal{S} \qquad \text{if no trellis termination}$$

▷ Parameter $\overline{\gamma}_i(s', s)$:

$$\overline{\gamma}_i(s', s) = \ln(p(\mathbf{y_i}|s_{i-1} = s', s_i = s)) + \ln(P(s_i = s|s_{i-1} = s'))$$

⇒ Metric calculated for each transition between states $s_{i-1} = s'$ and $s_i = s$ on the basis of the received symbol and the a priori information.

MAP algorithm:log MAP

- An optimal implementation in the logarithmic domain is possible.

$\Rightarrow$ Instead of approximation, use exact expression:

$$\begin{aligned} \ln(\exp(x)+\exp(y)) &= \max(x,y)+\ln(1+\exp(-|x-y|)) \\ &= \overset{*}{\max}(x,y) \end{aligned}$$

If more than two entries:

$$\begin{aligned} \ln(\exp(x)+\exp(y)+\exp(z)) &= \overset{*}{\max}(x,y,z) \\ &= \overset{*}{\max}(\overset{*}{\max}(x,y),z) \end{aligned}$$

$\Rightarrow$ Generalized maximum function.

MAP algorithm:log MAP

$\Rightarrow$ LOG-MAP algorithm.

- Proceeds exactly as the MAX-LOG-MAP algorithms

  if we replace every max function with a $\max^*$ function:

$$
\begin{aligned}
L_p(u_i) &= \max_{\mathcal{S}^+}{}^{*}(\overline{\alpha}_{i-1}(s') + \overline{\gamma}_i(s', s) + \overline{\beta}_i(s)) \\
&- \max_{\mathcal{S}^-}{}^{*}(\overline{\alpha}_{i-1}(s') + \overline{\gamma}_i(s', s) + \overline{\beta}_i(s))
\end{aligned}
$$

- Optimal algorithm!
- Numerical problems solved.
- 2 instances of a generalized VA.
- Complexity $\mathcal{O}(2^K)$ where K is the code constraint length.

Summary of algorithms

- Optimal algorithm: MAP.
  - ▷ Consider all paths in the trellis at each step.

    Divide them into 2 sets at step i.
- Optimal algorithm in the log. domain: LOG-MAP.
  - ▷ Consider all paths in the trellis at each step.

    Divide them into 2 sets at step i.
- Suboptimal algorithm in the log. domain: MAX-LOG-MAP.
  - ▷ Consider 2 paths per step:

    The best with bit 0 and the best with bit 1 at step i

Metric computation (1)

- Particular case: rate-1/2 RSC code.

  Notations already defined.

- Transition metric $\overline{\gamma}_i(s', s) = \ln(\gamma_i(s', s))$

  suited for LOG-MAP, MAX-LOG-MAP and SOVA algorithms.

- Metric $\overline{\gamma}_i(s', s)$ for a transition between states $s_{i-1} = s'$ and $s_i = s$:

$$\begin{aligned}\overline{\gamma}_i(s_{i-1} = s', s_i = s) &= \ln\left(\mathbf{p}(\mathbf{y_i}|s_{i-1} = s', s_i = s)\right) \\ &+ \ln\left(P(s_i = s|s_{i-1} = s')\right)\end{aligned}$$

  or equivalently, in terms of symbols:

$$\begin{aligned}\overline{\gamma}_i(s', s) &= \ln\left(p(\mathbf{y_i}|u_i, s_{i-1} = s')\right) + \ln\left(P(u_i)\right) \\ &= \ln\left(p(\mathbf{y_i}|u_i, u_{i-1}, \ldots, u_{i-M})\right) + \ln\left(P(u_i)\right) \\ &= \ln\left(p(\mathbf{y_i}|x_i^s, x_i^p)\right) + \ln\left(P(u_i)\right) = \ln\left(p(\mathbf{y_i}|b_i^s, b_i^p)\right) + \ln\left(P(u_i)\right)\end{aligned}$$

Metric computation (2)

▷ The first term $\ln(p(\mathbf{y}_i|b_i^s, b_i^p))$ depends on the received symbols.

Considering an AWGN channel with noise variance σ^2:

$$p(\mathbf{y}_i|b_i^s, b_i^p) = \frac{1}{\sigma^2 2\pi} \exp\left(-\frac{(y_i^s - b_i^s)^2 + (y_i^p - b_i^p)^2}{2\sigma^2}\right)$$

or, in the logarithmic domain:

$$\ln(p(\mathbf{y}_i|b_i^s, b_i^p)) = -\ln(\sigma^2 2\pi) - \frac{(y_i^s - b_i^s)^2 + (y_i^p - b_i^p)^2}{2\sigma^2}$$

which may be developed as:

$$\ln(p(\mathbf{y}_i|b_i^s, b_i^p)) = -\ln(\sigma^2 2\pi) + \frac{y_i^s b_i^s + y_i^p b_i^p}{\sigma^2} - \frac{(y_i^s)^2 + (b_i^s)^2 + (y_i^p)^2 + (b_i^p)^2}{2\sigma^2}$$

Metric computation (3)

▷ The second term $\ln(P(u_i))$ is calculated on the basis of the a priori information:

$$L_a(u_i) \approx \ln\left(\frac{P(u_i=1)}{P(u_i=0)}\right)$$

We may write:

$$P(u_i) = \begin{cases} \frac{\exp(L_a(u_i))}{1+\exp(L_a(u_i))} & \text{if } u_i = 1 \\ \frac{1}{1+\exp(L_a(u_i))} & \text{if } u_i = 0 \end{cases}$$

or, in the logarithmic domain:

$$\ln(P(u_i)) = L_a(u_i)u_i - \ln(1+\exp(L_a(u_i)))$$

Metric computation (4)

- Combining those two terms, we obtain:

$$\overline{\gamma}_i(s', s) = \begin{aligned} & - \quad \ln(\sigma^2 2\pi) + \frac{y_i^s b_i^s + y_i^p b_i^p}{\sigma^2} - \frac{(y_i^s)^2 + (b_i^s)^2 + (y_i^p)^2 + (b_i^p)^2}{2\sigma^2} \\ & + \quad L_a(u_i)u_i - \ln\left(1 + \exp(L_a(u_i))\right) \end{aligned}$$

- Suppressing the terms common to all hypotheses

  (terms which do not depend on u_i, b_i^s or b_i^p):

$$\overline{\gamma}_i(s', s) = \frac{y_i^s b_i^s + y_i^p b_i^p}{\sigma^2} + L_a(u_i)u_i$$

- Remembering that $L_c = \frac{2}{\sigma^2}$ for an AWGN channel:

$$\overline{\gamma}_i(s', s) = \frac{1}{2}(L_c y_i^s) b_i^s + \frac{1}{2}(L_c y_i^p) b_i^p + L_a(u_i)u_i$$

Metric computation (5)

- Noting that $b_i^s = 2x_i^s - 1$ and $b_i^p = 2x_i^p - 1$:

$$\overline{\gamma}_i(s', s) = (L_c y_i^s) x_i^s + (L_c y_i^p) x_i^p + L_a(u_i) u_i$$

- Remembering that $x_i^s = u_i$:

$$\overline{\gamma}_i(s', s) = (L_c y_i^s + L_a(u_i)) u_i + (L_c y_i^p) x_i^p$$

$\Rightarrow$ For each transition $(s_{i-1} = s', s_i = s)$ in the trellis (characterized by u_i, $x_i^s = u_i$ and x_i^p), we can compute the metric on the basis of the a priori information $L_a(u_i)$ and the soft outputs of the channel $L_c y_i^s$ and $L_c y_i^p$.

Metric : fundamental property

- SISO decoder fundamental property for a rate-1/2 RSC code:

$$L_p(u_i) = L_c y_i^s + L_a(u_i) + L_e(u_i)$$

- Expression of the transition metric:

$$\begin{aligned}\gamma_i(s', s) &= \exp\left(\overline{\gamma}_i(s', s)\right) \\ &= \exp((L_c y_i^s + L_a(u_i))u_i + L_c y_i^p x_i^p)\end{aligned}$$

can be written as:

$$\gamma_i(s', s) = \exp((L_c y_i^s + L_a(u_i))u_i)\gamma_i^e(s', s)$$

with:

$$\gamma_i^e(s', s) = \exp\left(L_c y_i^p x_i^p\right)$$

Metric : fundamental property

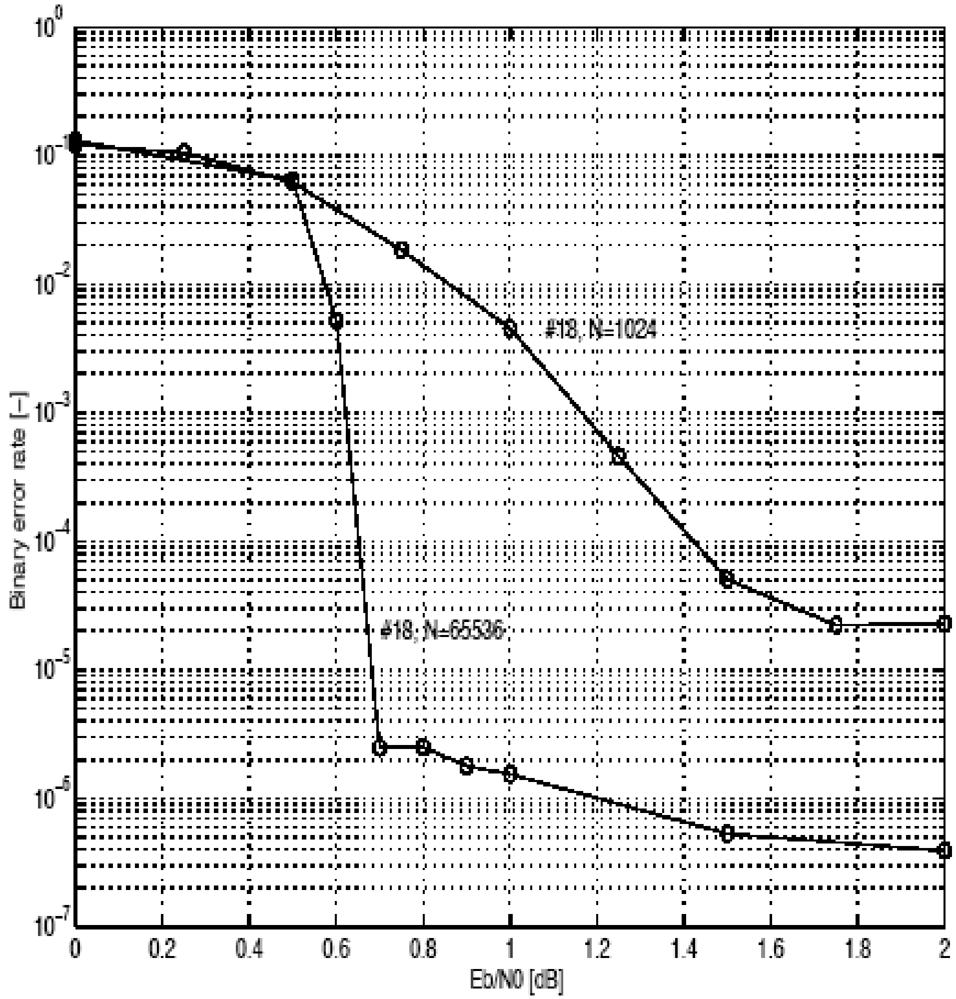
- According to the MAP algorithm:

$$
\begin{aligned}
L_p(u_i) &= \ln\left(\frac{P(u_i=1|\mathbf{y})}{P(u_i=0|\mathbf{y})}\right) \\
&= \ln\left(\frac{\sum_{\mathcal{S}^+}\alpha_{i-1}(s').\gamma_i(s',s).\beta_i(s)}{\sum_{\mathcal{S}^-}\alpha_{i-1}(s').\gamma_i(s',s).\beta_i(s)}\right) \\
&= \ln\left(\frac{\sum_{\mathcal{S}^+}\alpha_{i-1}(s').\exp((L_c y_i^s + L_a(u_i))u_i)\gamma_i^e(s',s).\beta_i(s)}{\sum_{\mathcal{S}^-}\alpha_{i-1}(s').\exp((L_c y_i^s + L_a(u_i))u_i)\gamma_i^e(s',s).\beta_i(s)}\right)
\end{aligned}
$$

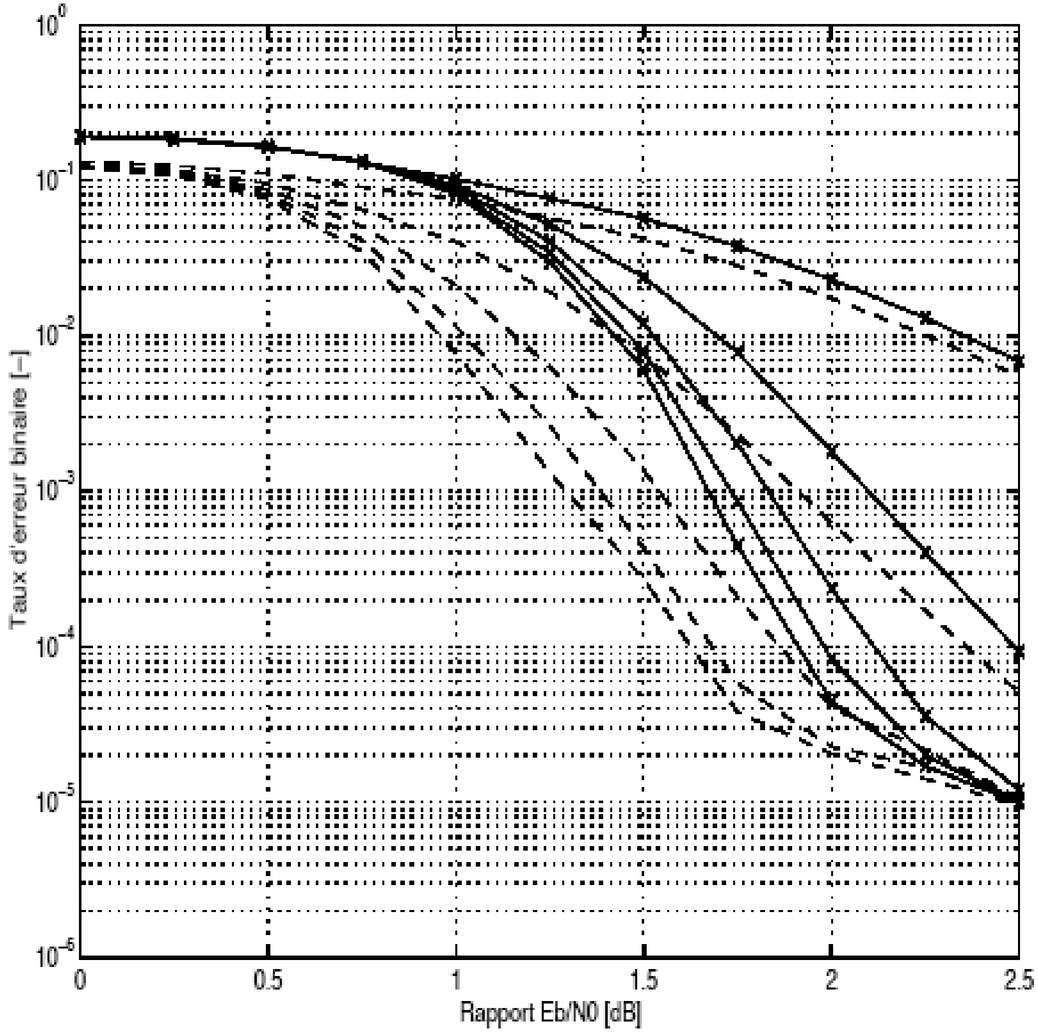
- Factors $\exp((L_c y_i^s + L_a(u_i))u_i)$ identical for all transitions in $\mathcal{S}^+$ and $\mathcal{S}^-$ $\Rightarrow$

$$
\begin{aligned}
L_p(u_i) &= \ln\left(\frac{\exp((L_c y_i^s + L_a(u_i)).1)\sum_{\mathcal{S}^+}\alpha_{i-1}(s').\gamma_i^e(s',s).\beta_i(s)}{\exp((L_c y_i^s + L_a(u_i)).0)\sum_{\mathcal{S}^-}\alpha_{i-1}(s').\gamma_i^e(s',s).\beta_i(s)}\right) \\
&= L_c y_i^s + L_a(u_i) + \ln\left(\frac{\sum_{\mathcal{S}^+}\alpha_{i-1}(s').\gamma_i^e(s',s).\beta_i(s)}{\sum_{\mathcal{S}^-}\alpha_{i-1}(s').\gamma_i^e(s',s).\beta_i(s)}\right) \\
&= L_c y_i^s + L_a(u_i) + L_e(u_i)
\end{aligned}
$$

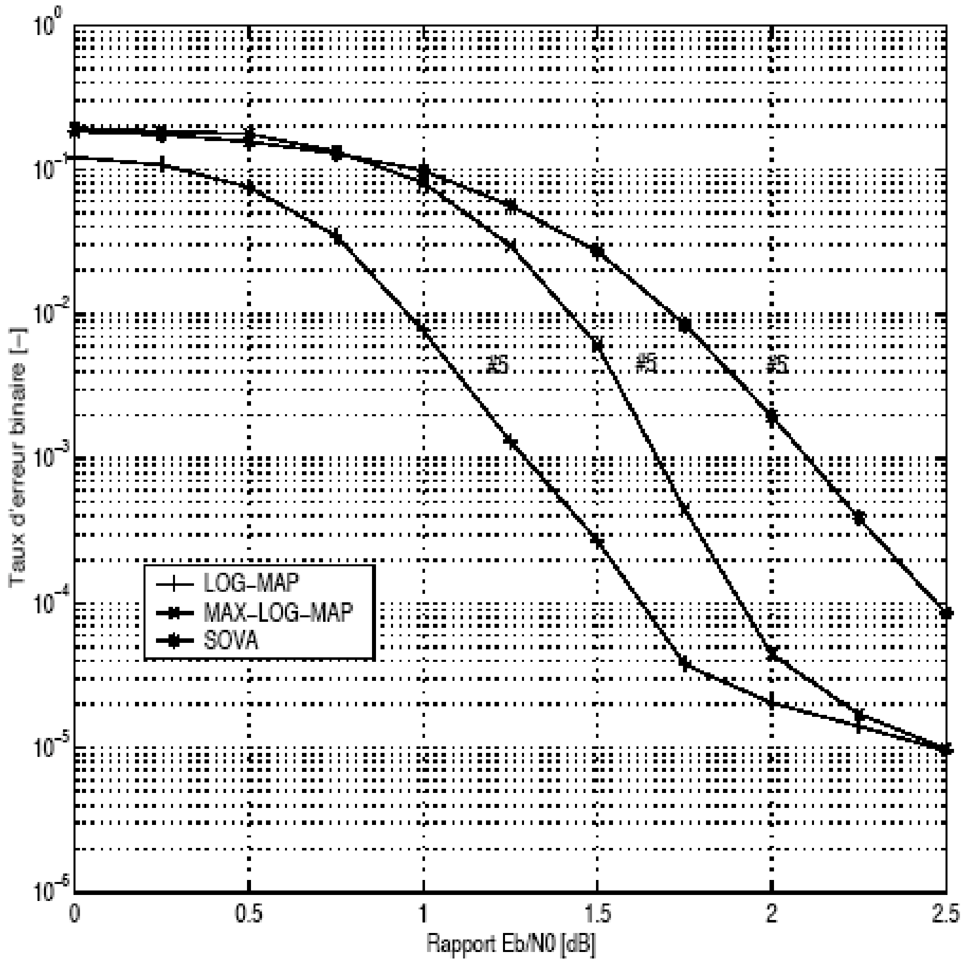
Impact of interleaver



Log MAP vs MAX LOG MAP



Comparison

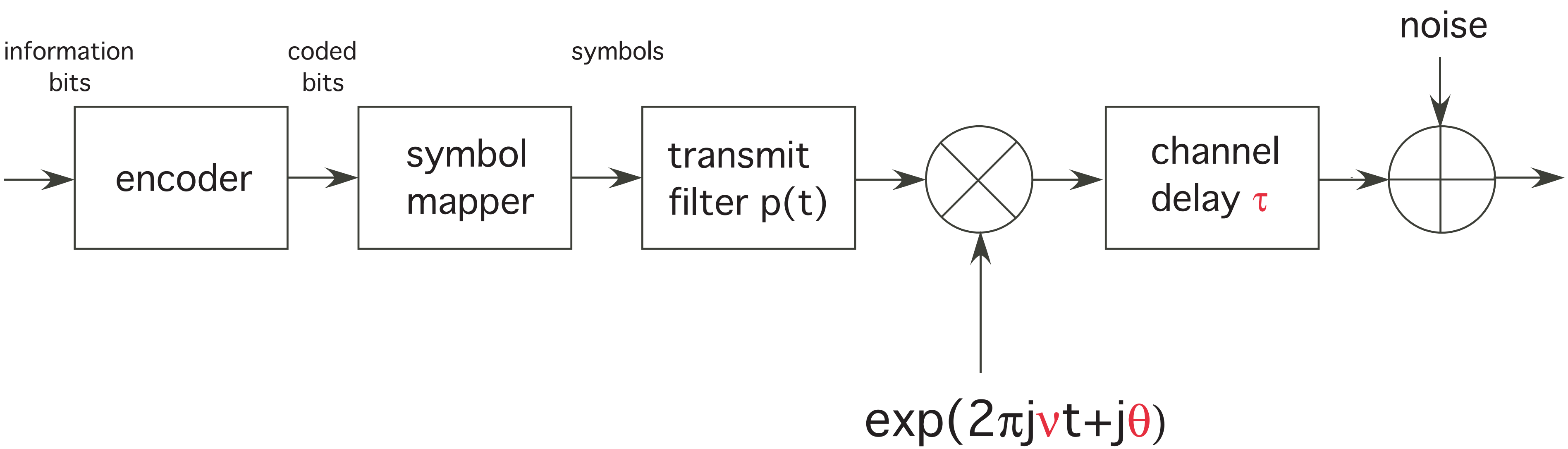


Taux d'erreur binaire [–]
Rapport Eb/N0 [dB]
LOG–MAP
MAX–LOG–MAP
SOVA

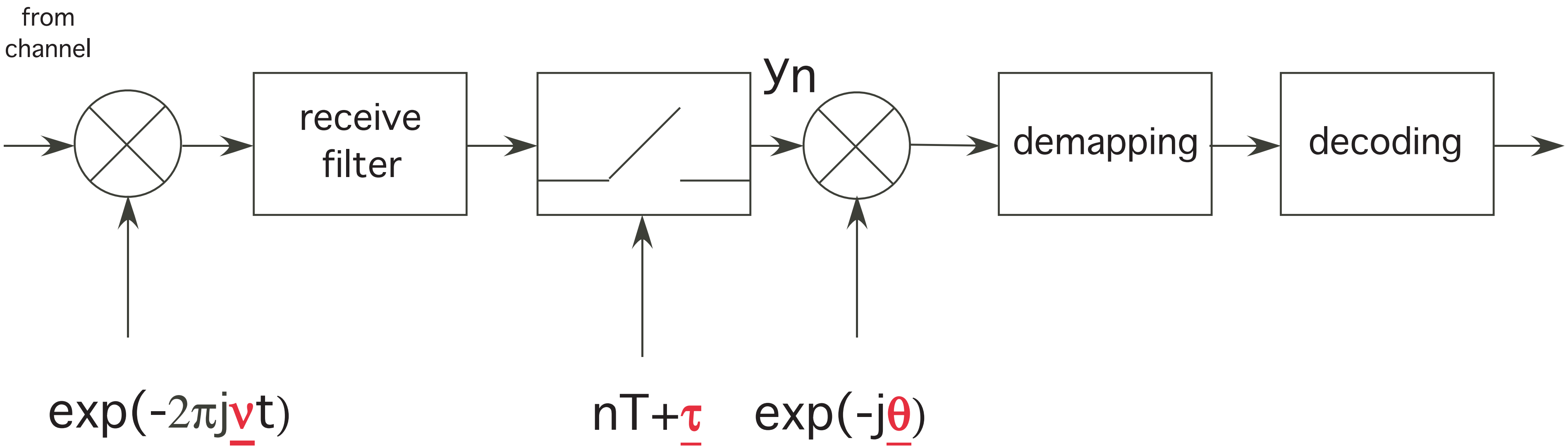
Outline

- Introduction/motivation
- The EM algorithm
- Coding and the MAP algorithm
- **Synchronization of coded systems with the EM algorithm**
- Illustration of performance
- CSI estimation for coded MIMO transmission
- Illustration and performance
- Cramer-Rao bound with coded/prior information

Transmitter setup



Receiver setup



Observation model

- Received signal

$$r(t) = \mathbf{A} \sum_{k=0}^{K-1} a_k \, p(t - kT - \tau) \; e^{j(2\pi\nu t + \theta)} + w(t), \qquad (22)$$

- A: amplitude; τ: timing; (ν, θ) carrier frequency and phase offset
- $w(t)$ AWGN
- a_k data symbols

EM algorithm

- It comes

$$\ln p(\mathbf{r}|\mathbf{a},\tilde{\mathbf{b}}) = -2\tilde{A}\ \mathrm{Re}\{\sum_{k=0}^{K-1} a_k^*\, y_k(\tilde{\nu},\tilde{\tau})\, e^{-j\tilde{\theta}}\} + \tilde{A}^2 \sum_{k=0}^{K-1} |a_k|^2, \quad (23)$$

where

$$y_k(\tilde{\nu},\tilde{\tau}) \triangleq \int_{-\infty}^{+\infty} r(t)\, e^{-j(2\pi\tilde{\nu}t)}\, p(t-kT-\tilde{\tau})\, dt. \quad (24)$$

Posterior averages

- Expectation step:

$$\mathcal{Q}(\tilde{\mathbf{b}}, \hat{\mathbf{b}}^{(n-1)}) = -2\tilde{A}\ \mathrm{Re}\{\sum_{k=0}^{K-1} \eta_k^*(\mathbf{r}, \hat{\mathbf{b}}^{(n-1)})\, y_k(\tilde{\nu}, \tilde{\tau})\, e^{-j\tilde{\theta}}\} + \tilde{A}^2 \sum_{k=0}^{K-1} \rho_k(\mathbf{r}, \hat{\mathbf{b}}^{(n-1)}). \quad (25)$$

- With following posterior values

$$\eta_k(\mathbf{r}, \hat{\mathbf{b}}^{(n-1)}) \triangleq \int_{\mathbf{a}\in\mathcal{A}^K} a(k)\ p(\mathbf{a}|\mathbf{r}, \hat{\mathbf{b}}^{(n-1)})\, d\mathbf{a}$$

$$\rho_k(\mathbf{r}, \hat{\mathbf{b}}^{(n-1)}) \triangleq \int_{\mathbf{a}\in\mathcal{A}^K} |a(k)|^2\, p(\mathbf{a}|\mathbf{r}, \hat{\mathbf{b}}^{(n-1)})\, d\mathbf{a}$$

- Note: depend on symbol marginal posterior probabilities !

EM estimates

- Maximization step leads to partially decoupled solutions [ICC2003]

$$[\hat{\nu}^{(n)}, \hat{\tau}^{(n)}] = \arg\max_{\tilde{\nu},\tilde{\tau}} \{ | \sum_{k=0}^{K-1} \eta_{\mathbf{k}}^{*}(\mathbf{r}, \hat{\mathbf{b}}^{(\mathbf{n-1})})\, y_k(\tilde{\nu}, \tilde{\tau}) | \} \qquad (26)$$

$$\hat{\theta}^{(n)} = \arg\{ \sum_{k=0}^{K-1} \eta_{\mathbf{k}}^{*}(\mathbf{r}, \hat{\mathbf{b}}^{(\mathbf{n-1})})\, y_k(\hat{\nu}^{(n)}, \hat{\tau}^{(n)}) \} \qquad (27)$$

$$\hat{A}^{(n)} = \frac{| \sum_{k=0}^{K-1} \eta_{\mathbf{k}}^{*}(\mathbf{r}, \hat{\mathbf{b}}^{(\mathbf{n-1})})\, y_k(\hat{\nu}^{(n)}, \hat{\tau}^{(n)}) |}{\sum_{k=0}^{K-1} \rho_{\mathbf{k}}(\mathbf{r}, \hat{\mathbf{b}}^{(\mathbf{n-1})})}. \qquad (28)$$

Comparison with pilot aided solution

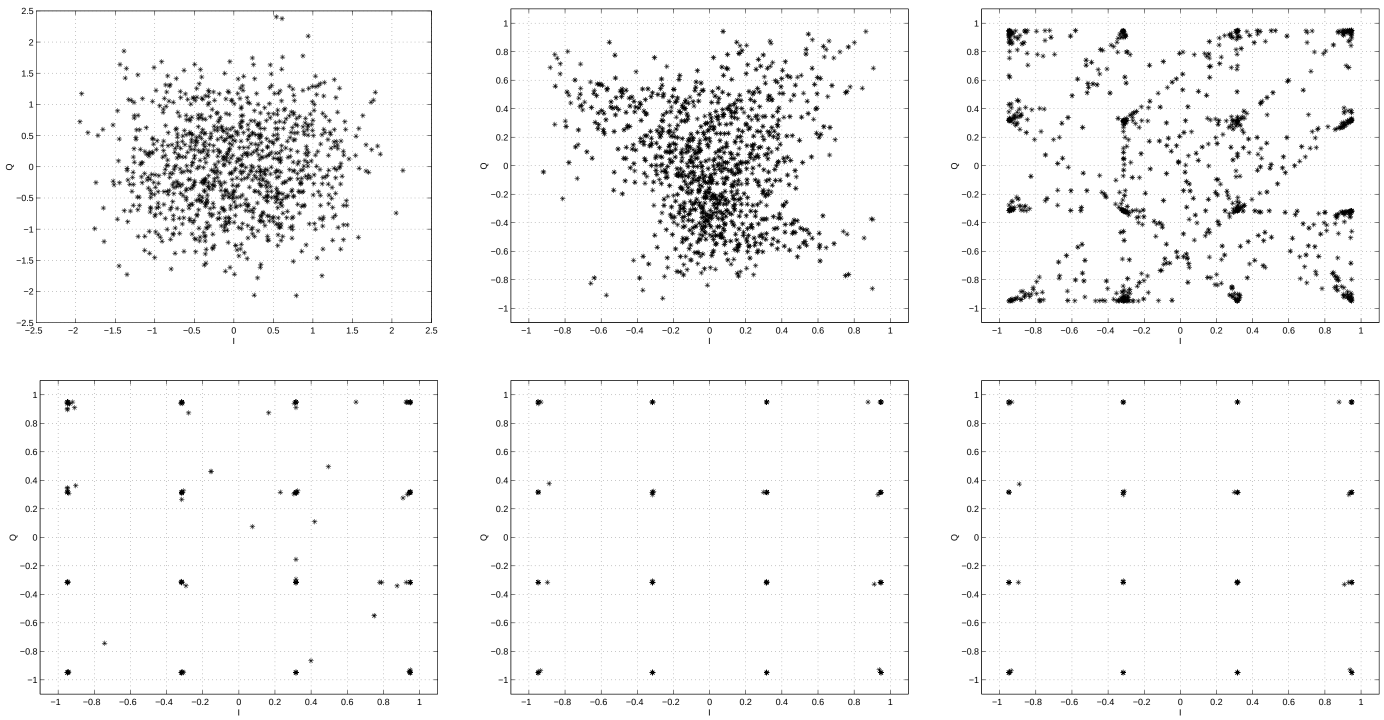
- If pilots had been used

$$[\hat{\nu}, \hat{\tau}] = \arg\max_{\tilde{\nu},\tilde{\tau}} \{ | \sum_{k=0}^{K-1} \mathbf{a_k^*} \, y_k(\tilde{\nu}, \tilde{\tau}) | \} \tag{29}$$

$$\hat{\theta} = \arg\{ \sum_{k=0}^{K-1} \mathbf{a_k^*} \, y_k(\hat{\nu}, \hat{\tau}) \} \tag{30}$$

$$\hat{A}^{(n)} = \frac{| \sum_{k=0}^{K-1} \mathbf{a_k^*} \, y_k(\hat{\nu}, \hat{\tau}) |}{\sum_{k=0}^{K-1} |\mathbf{a_k}|^2}. \tag{31}$$

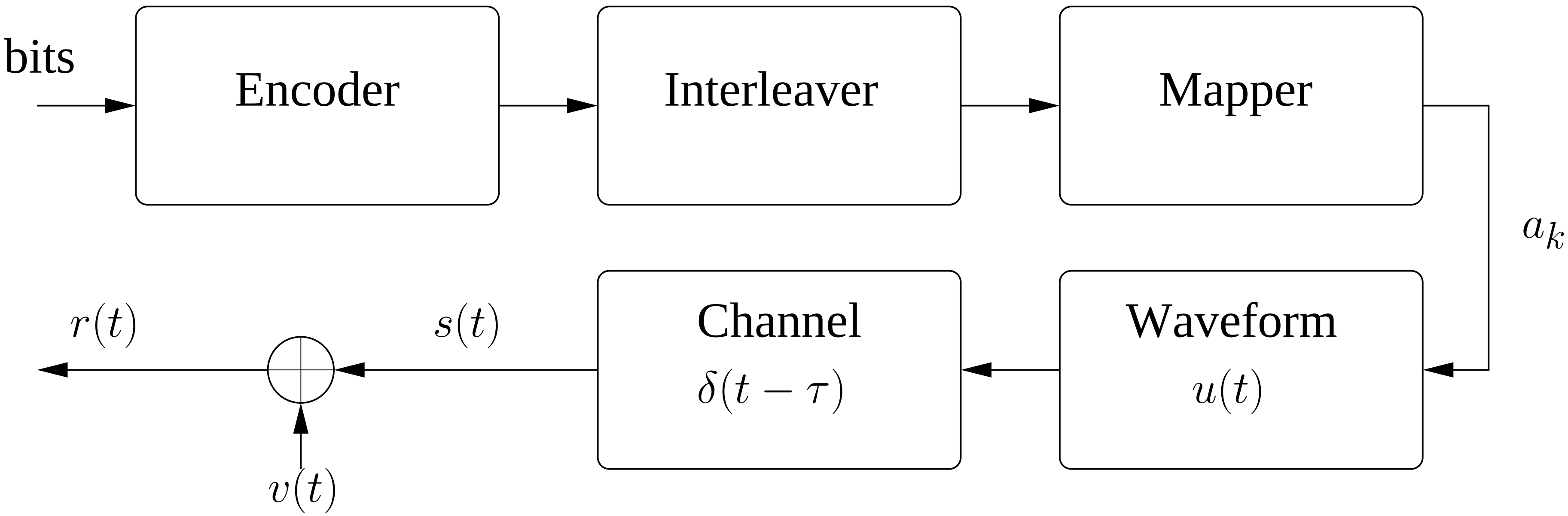
Posterior mean values



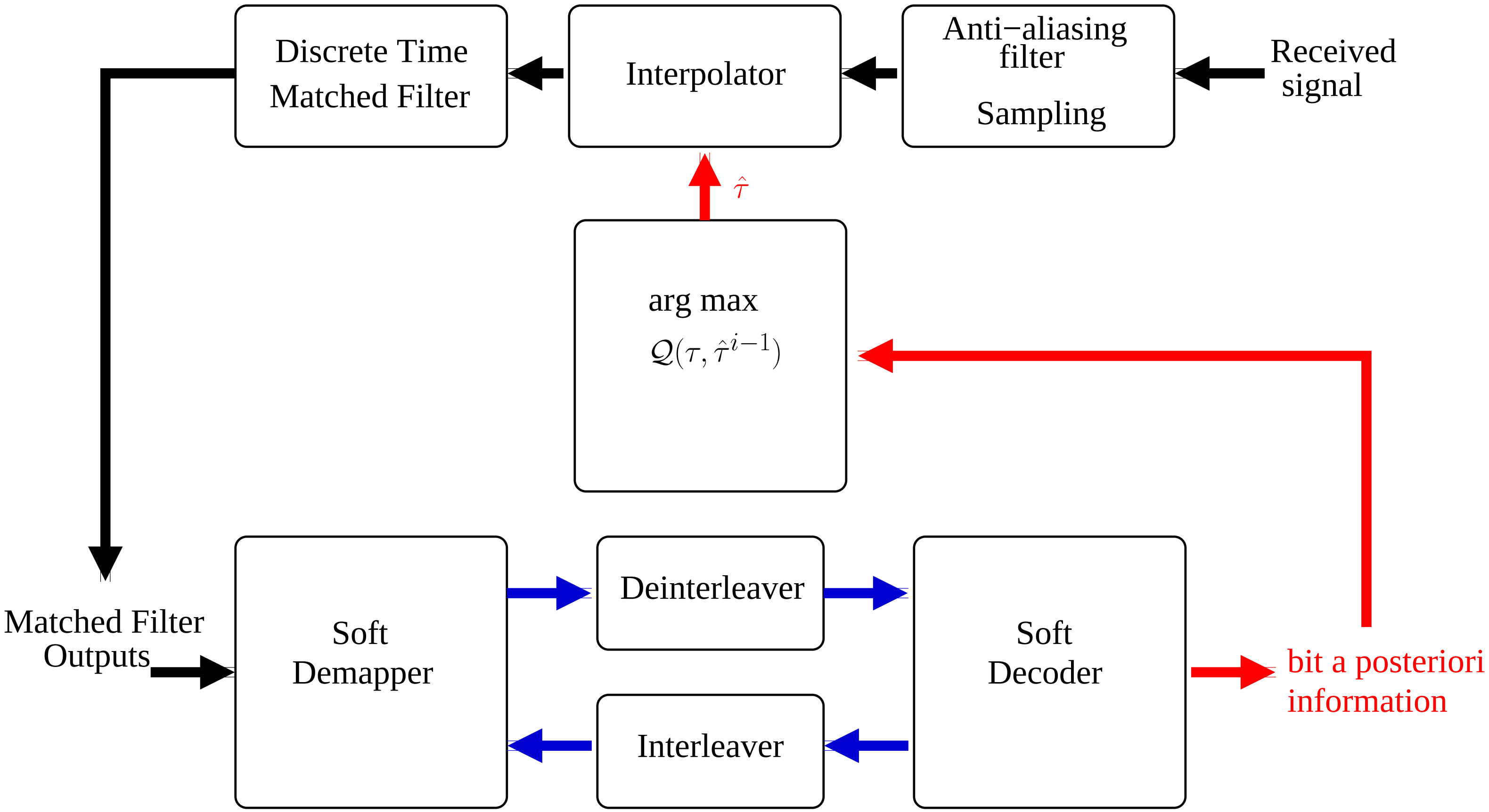
Discussion

- Solution only requires marginal symbol a posteriori probabilities
- Delivered by trellis based MAP module implemented by means of BCJR algorithm (when code or *supercode* not too complex)
- Also available in a turbo receiver after *sufficient* number of iterations

BICM transmitter



BICM iterative demapper/decoder with timing estimation



Discussion

- A turbo receiver is supposed to deliver bit posterior probabilities after an infinite number of iterations
- Approximation: use these bit APPs obtained after one or several iterations to build symbol APPS
- Use them in the EM algorithm

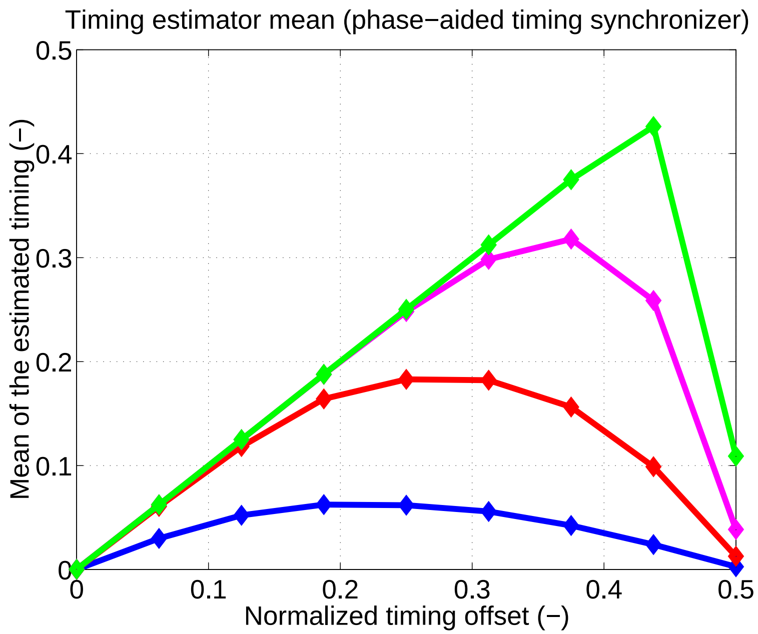
Outline

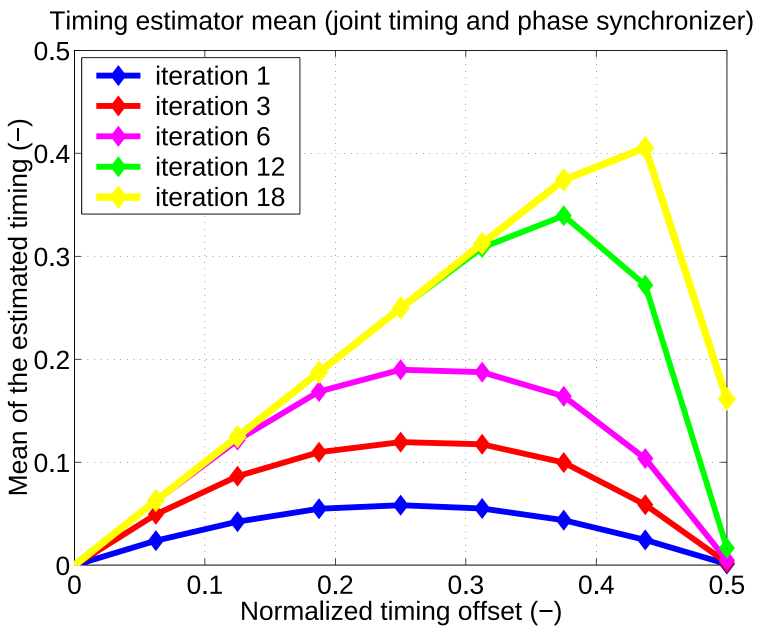
- Introduction/motivation
- The EM algorithm
- Coding and the MAP algorithm
- Synchronization of coded systems with the EM algorithm
- **Illustration of performance**
- CSI estimation for coded MIMO transmission
- Illustration and performance
- Cramer-Rao bound with coded/prior information

Setup

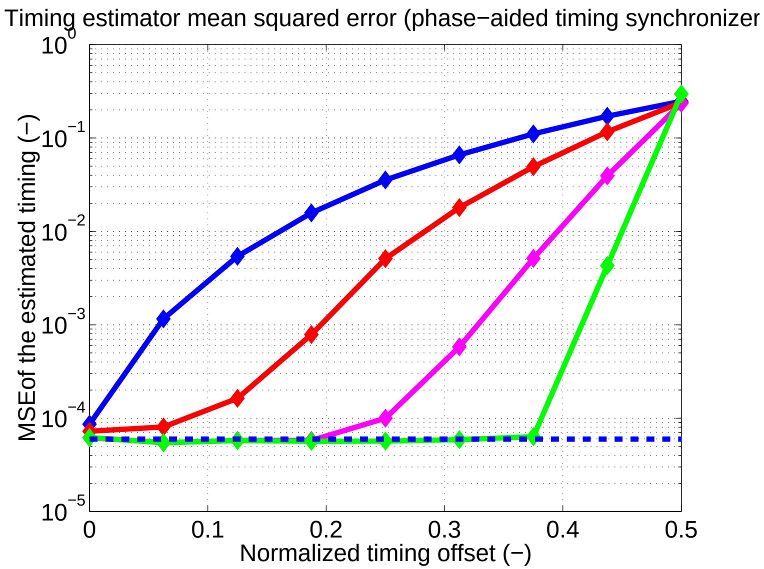
- 16-QAM, "medium unconditioned bit-wise mutual information" mapping, convolutional code, length$= 3$, code rate$= 1/2$
- Timing only or joint phase/timing estimation
- Startup : $\hat{\tau}^{(0)} = 0$ or $\hat{\tau}^{(0)} = 0$,$\hat{\theta}^{(0)} = 0$ ($\theta = 15$ degrees)
- $E_b/N_0 = 4$dB
- One turbo iteration per EM iteration (no reset of extrinsic information)

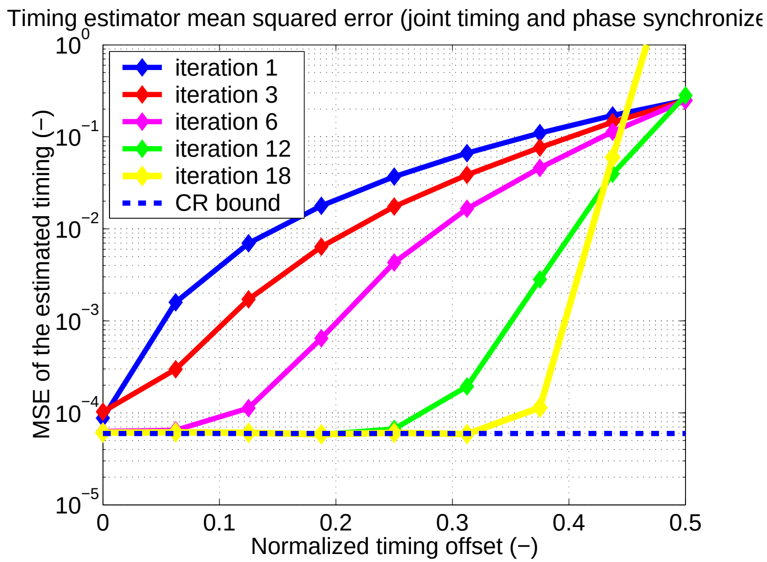
Results: mean



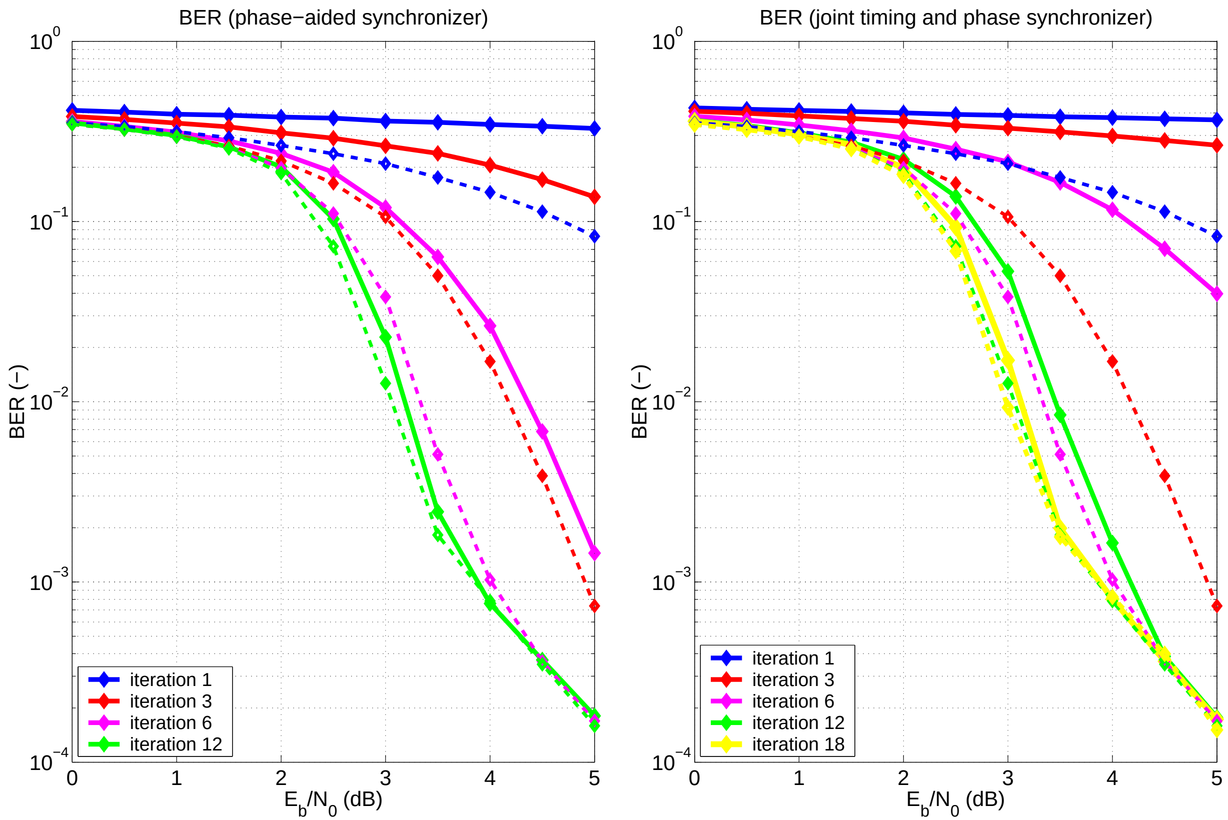


Results: MSE


Timing estimator mean squared error (phase−aided timing synchronizer
MSEof the estimated timing (−)
Normalized timing offset (−)
10^0
10^−1
10^−2
10^−3
10^−4
10^−5
0
0.1
0.2
0.3
0.4
0.5


Timing estimator mean squared error (joint timing and phase synchronize
MSE of the estimated timing (−)
Normalized timing offset (−)
iteration 1
iteration 3
iteration 6
iteration 12
iteration 18
CR bound
10^0
10^−1
10^−2
10^−3
10^−4
10^−5
0
0.1
0.2
0.3
0.4
0.5

Results: BER ($\tau/T = 0.25$)



Steepest descent implementation

- No closed form solution for the symbol timing
- Steepest descent leads to

$$\hat{\epsilon}^{(n)} \triangleq \hat{\tau}^{(n+1)} - \hat{\tau}^{(n)} = \beta \sum_k |\eta_k^{(n)}| \times \text{Re}\{e^{-j\arg(\eta_k^{(n)})} \dot{y}(kT + \hat{\tau}^{(n)})\} \quad (32)$$

- Proposal: design a best linear unbiased estimator [SPAWC2003]

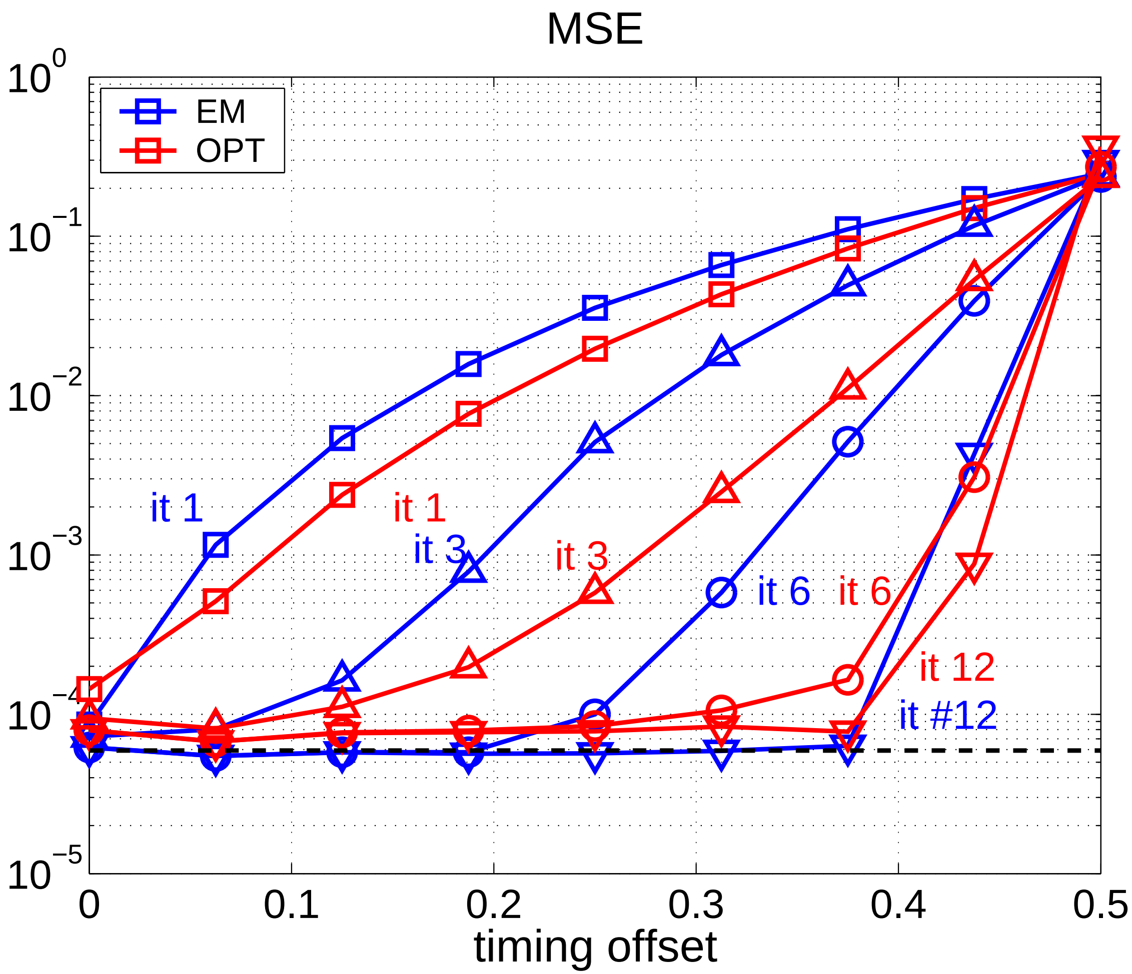
BLUE estimator

- BLUE estimator (with some simplification) leads to

$$\hat{\epsilon}^{(n)} = \beta' \sum_k \frac{\mathsf{E}[h_I(k)]}{\sigma^2_{w_I(k)} + \sigma^2_{e_I(k)}} \times \mathrm{Re}\{ e^{-j\arg(\eta_k^{(n)})} (\dot{y}(kT + \hat{\tau}^{(n)}) - \sum_{\mathbf{k}'} \eta_{\mathbf{k}'}^{(\mathbf{n})} \dot{\mathbf{x}}_{\mathbf{k}-\mathbf{k}'})\}$$
$$+ \beta' \sum_{\mathbf{k}} \frac{\mathsf{E}[\mathbf{h_Q}(\mathbf{k})]}{\sigma^2_{\mathbf{w_Q}(\mathbf{k})} + \sigma^2_{\mathbf{e_Q}(\mathbf{k})}} \times \mathrm{Im}\{ \mathbf{e}^{-\mathbf{j}\arg(\eta_{\mathbf{k}}^{(\mathbf{n})})} (\dot{\mathbf{y}}(\mathbf{kT} + \hat{\tau}^{(\mathbf{n})}) - \sum_{\mathbf{k}'} \eta_{\mathbf{k}'}^{(\mathbf{n})} \dot{\mathbf{x}}_{\mathbf{k}-\mathbf{k}'})\}$$

- Idea: not only projection in phase with $\eta_k^{(n)}$ contains useful information but also that in quadrature (**red** term).
- Also: perform soft interference cancellation of self noise (**blue** term)

Results with improved design



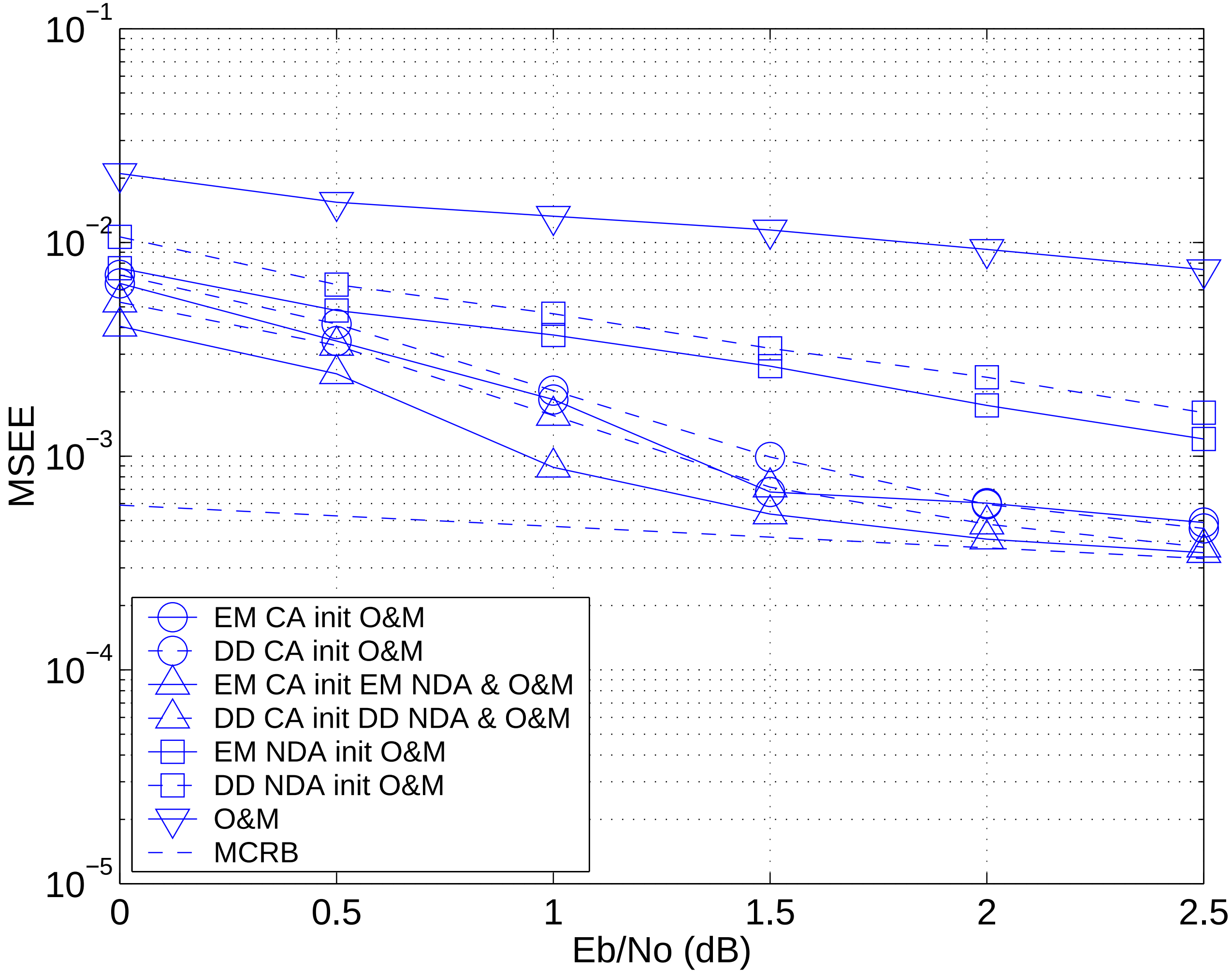
Acquisition

- Does not solve acquisition
- Conventional methods with ambiguity resolution can be used to initialize the EM estimates.
- Or run the EM with different initial values [Wymeersch2004] . Can work without pilots at low SNRs ((M)CRB reached at 1dB).
- Solves convergence towards local minimum.

Turbo coded system

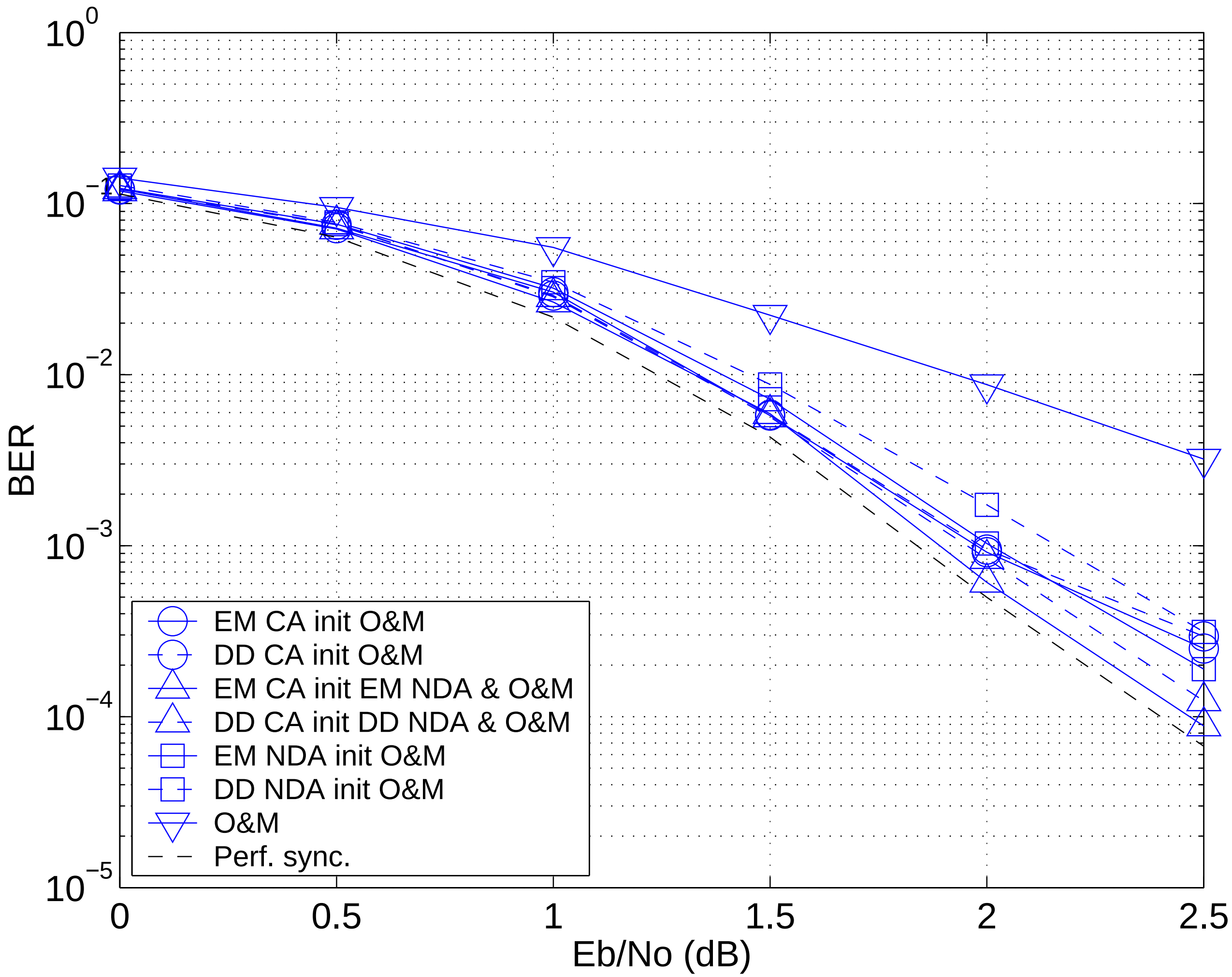
- 512 BPSK symbols
- Timing changed randomly at each new frame
- MSE and BER with different initial values for the EM

MSE results



10^−1
10^−2
10^−3
10^−4
10^−5
MSEE
0
0.5
1
1.5
2
2.5
Eb/No (dB)
EM CA init O&M
DD CA init O&M
EM CA init EM NDA & O&M
DD CA init DD NDA & O&M
EM NDA init O&M
DD NDA init O&M
O&M
MCRB

BER results



BER
Eb/No (dB)
10^0
10^{-1}
10^{-2}
10^{-3}
10^{-4}
10^{-5}
0
0.5
1
1.5
2
2.5
EM CA init O&M
DD CA init O&M
EM CA init EM NDA & O&M
DD CA init DD NDA & O&M
EM NDA init O&M
DD NDA init O&M
O&M
Perf. sync.

Conclusion

- Soft data aided synchronization works
- Cramér Rao bound can be reached
- Initial value has large impact

Outline

- Introduction/motivation
- The EM algorithm
- Coding and the MAP algorithm
- Synchronization of coded systems with the EM algorithm
- Illustration of performance
- **CSI estimation for coded MIMO transmission**
- Illustration and performance
- Cramer-Rao bound with coded/prior information

FS MIMO scheme

- FS MIMO channels with n_t transmit and n_r receive antennas
- Observation model for polyphase component m and RX antenna j

$$\underline{r}_m^{(j)} = \underline{\underline{A}}\,\underline{h}_m^{(j)} + \underline{n}_m^{(j)}. \tag{33}$$

- Objective: estimate the $\underline{h}_m^{(j)}$; the symbols $a_i(m)$ are nuisance parameters
- Estimation of noise variance can be handled as well

EM algorithm

- Follow path similar to soft data aided synchronization [Wautelet2003]

$$\ln p(\mathcal{R}|\underline{\underline{A}}, \tilde{\mathcal{B}}) = -\frac{1}{\tilde{\sigma}_n^2} \sum_{j=1}^{n_R} \sum_{m=0}^{M_s-1} (\underline{r}_m^{(j)} - \underline{\underline{A}}\, \underline{\tilde{h}}_m^{(j)})^H (\underline{r}_m^{(j)} - \underline{\underline{A}}\, \underline{\tilde{h}}_m^{(j)}) \qquad (34)$$

- Channel estimation at step (n)

$$\underline{\hat{h}}_{m,\mathrm{EM}}^{(j)(n)} = E[\underline{\underline{A}}^H \underline{\underline{A}} | \mathcal{R}, \hat{\mathcal{B}}^{(n-1)}]^{-1} E[\underline{\underline{A}} | \mathcal{R}, \hat{\mathcal{B}}^{(n-1)}]^H \underline{r}_m^{(j)} \qquad (35)$$

- Noise-variance estimation

$$\hat{\sigma}_{n,\mathrm{EM}}^{2\,(n)} = \frac{1}{n_R M_s L_r} \sum_{j=1}^{n_R} \sum_{m=0}^{M_s-1} \Big[\underline{r}_m^{(j)\,H} \underline{r}_m^{(j)} + \underline{\hat{h}}_{m,\mathrm{EM}}^{(j)\,H} E[\underline{\underline{A}}^H \underline{\underline{A}} | \mathcal{R}, \hat{\mathcal{B}}^{(n-1)}]\, \underline{\hat{h}}_{m,\mathrm{EM}}^{(j)}$$
$$-2\mathrm{Re}\left\{ \underline{r}_m^{(j)\,H} E[\underline{\underline{A}} | \mathcal{R}, \hat{\mathcal{B}}^{(n-1)}]\, \underline{\hat{h}}_{m,\mathrm{EM}}^{(j)} \right\} \Big] .$$

Comparison with pilot aided solution

- Pilot aided solution for the channel

$$\underline{\hat{h}}^{(j)}_{m,\mathrm{DA}} = \left(\underline{\underline{A_p}}^H \underline{\underline{A_p}}\right)^{-1} \underline{\underline{A_p}}^H \underline{r_p}^{(j)}_m. \tag{36}$$

- For the noise-variance (biased):

$$\hat{\sigma}^2_{n,\mathrm{DA}} = \frac{1}{n_R M_s L_r} \sum_{j=1}^{n_R} \sum_{m=0}^{M_s-1} (\underline{r_p}^{(j)}_m - \underline{\underline{A_p}}\, \underline{\hat{h}}^{(i,j)}_{m,\mathrm{DA}})^H (\underline{r_p}^{(j)}_m - \underline{\underline{A_p}}\, \underline{\hat{h}}^{(i,j)}_{m,\mathrm{DA}}). \tag{37}$$

- Biased can be removed
- EM Channel estimation at step (n)

$$\underline{\hat{h}}^{(j)(n)}_{m,\mathrm{EM}} = E[\underline{\underline{A}}^H \underline{\underline{A}} | \mathcal{R}, \hat{\mathcal{B}}^{(n-1)}]^{-1} E[\underline{\underline{A}} | \mathcal{R}, \hat{\mathcal{B}}^{(n-1)}]^H \underline{r}^{(j)}_m \tag{38}$$

- Posterior averages of products also needed

Problems

- Posterior average of product not delivered by e. g. turbo receivers
- Solution for the channel estimate delivered at each EM iteration is **biased**
  - Degrades the BER
  - Pointed out by [Kobayashi et al.,2001]; ad-hoc solutions proposed
- Solution for the noise variance estimate delivered at each EM iteration is also **biased**: bias can be partly removed.

Proposed solution: BLUE design

- Target Best Linear Unbiased Estimator assuming a priori information for the symbols
- Estimation at step (n):

$$\underline{\hat{h}}^{(j)}_{m,\mathrm{UEM}} = (E[\underline{\underline{A}}|\mathcal{R},\hat{\mathcal{B}}^{(n-1)}]^H E[\underline{\underline{A}}|\mathcal{R},\hat{\mathcal{B}}^{(n-1)}])^{-1}\, E[\underline{\underline{A}}|\mathcal{R},\hat{\mathcal{B}}^{(n-1)}]^H\, \underline{r}^{(j)}_m. \qquad (39)$$

Other possibility: ECM

- Expectation Conditional Maximization
- Update one value at a time; take the most recent value for others
- Avoid matrix inversion

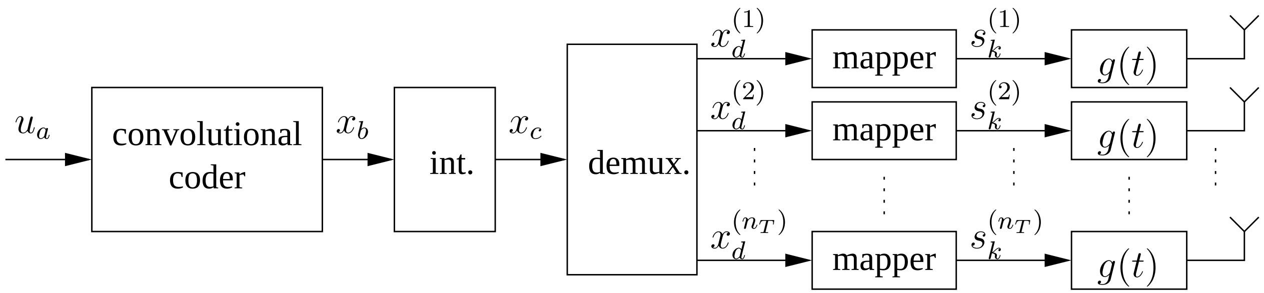
$$\hat{h}^{(i,j)(n)}_{l,m,\mathrm{ECM}} = \frac{\{E[\underline{\underline{S}}|\mathcal{R},\hat{\mathcal{B}}^{(n-1)}]^H\,\underline{r}^{(j)}_m\}_{Li+l} - \{E[\underline{\underline{S}}^H\underline{\underline{S}}|\mathcal{R},\hat{\mathcal{B}}^{(n-1)}]\,\underline{\tilde{h}}^{(i,j)(n)}_{l,m}\}_{Li+l}}{\{E[\underline{\underline{S}}^H\underline{\underline{S}}|\mathcal{R},\hat{\mathcal{B}}^{(n-1)}]\}_{Li+l,Li+l}}, \tag{40}$$

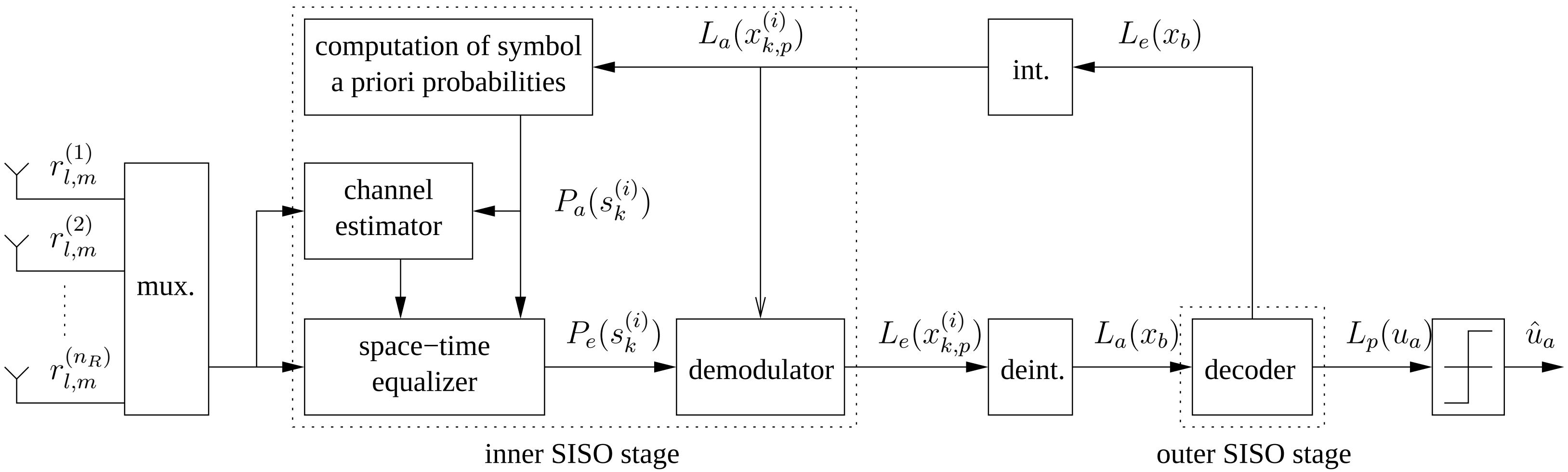
- This solution is also biased and the bias can be removed

Outline

- Introduction/motivation
- The EM algorithm
- Coding and the MAP algorithm
- Synchronization of coded systems with the EM algorithm
- Illustration of performance
- CSI estimation for coded MIMO transmission
- **Illustration and performance**
- Cramer-Rao bound with coded/prior information

ST BICM Transmitter and receiver

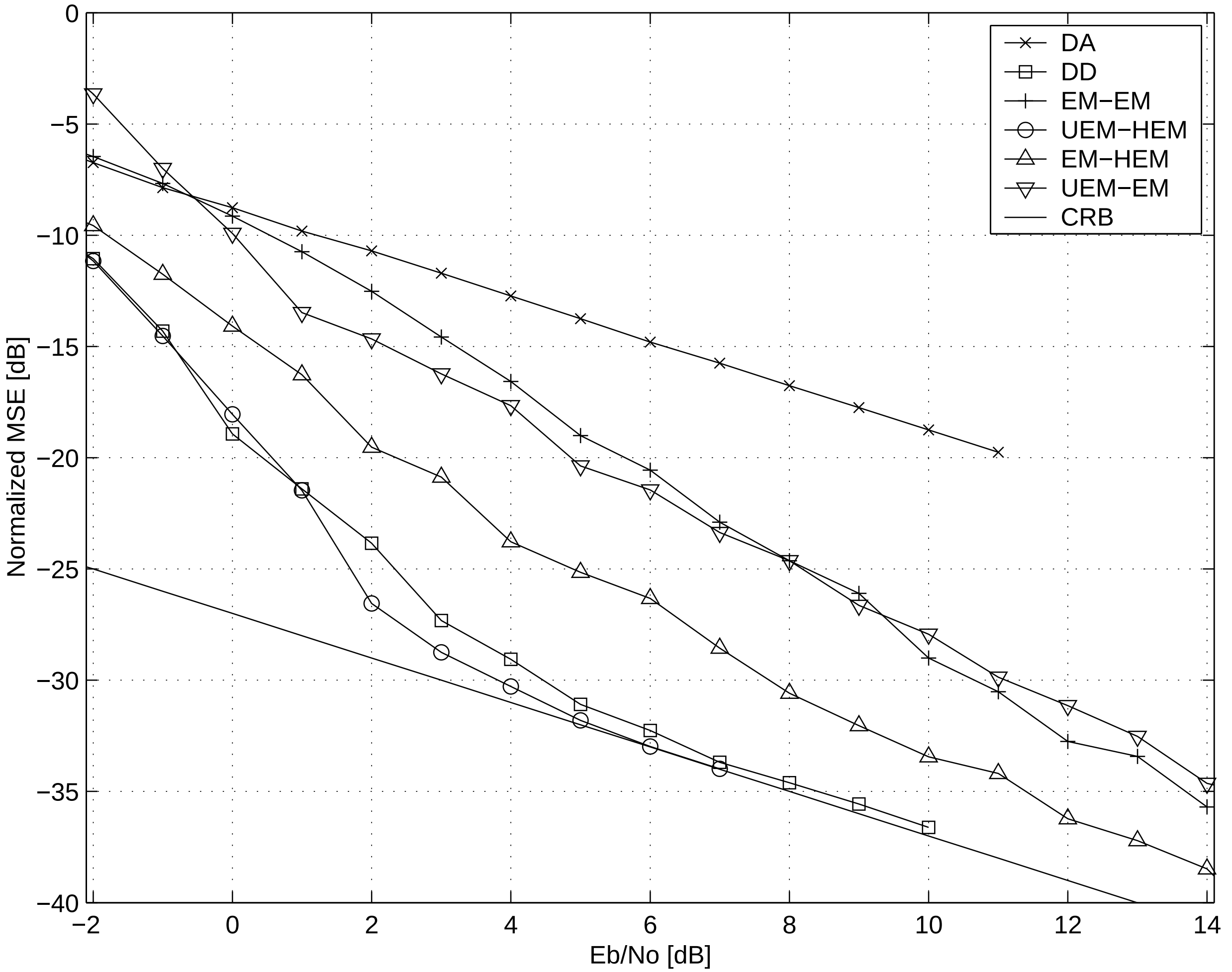




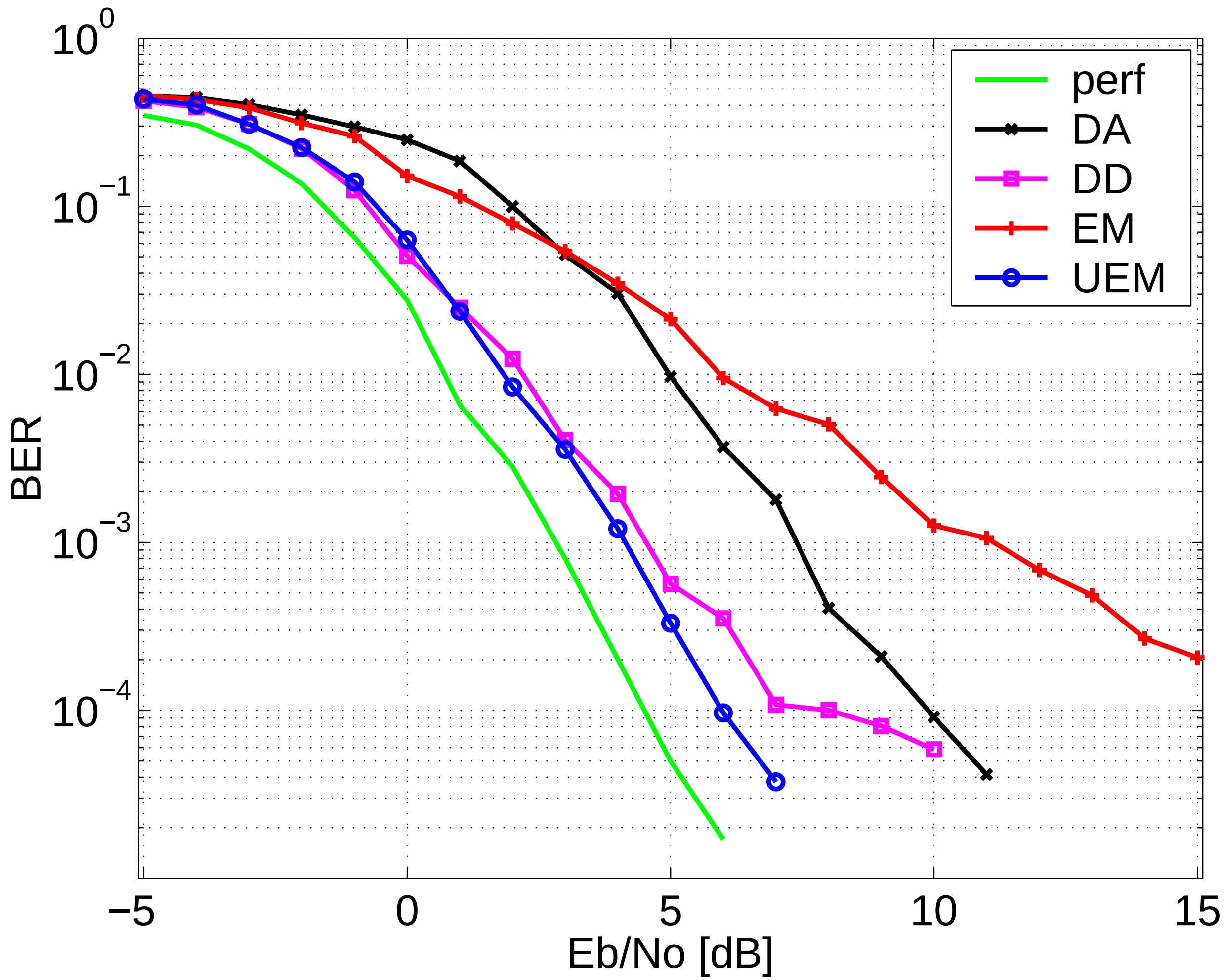
Simulation parameters

- Space time BICM [Tonello 2000]
- Random interleaver, 8-PSK, Gray Mapping
- $r = 0.5$ convolutional encoder, generator polynomials (23,35) (octal)
- frame: 2000 information bits (1336 symboles)
- Flat Rayleigh fading channel 4×4; 4×5 pilot symbols (orthogonal)
- FS GSM Typical Urban 4×4; 4×55 pilot symbols
- Iterative space equalization/demodulation (MMSE filter based) and decoding (BCJR) [Wautelet 2004]
- 6 iterations
- Noise variance estimated in a way similar to CSI

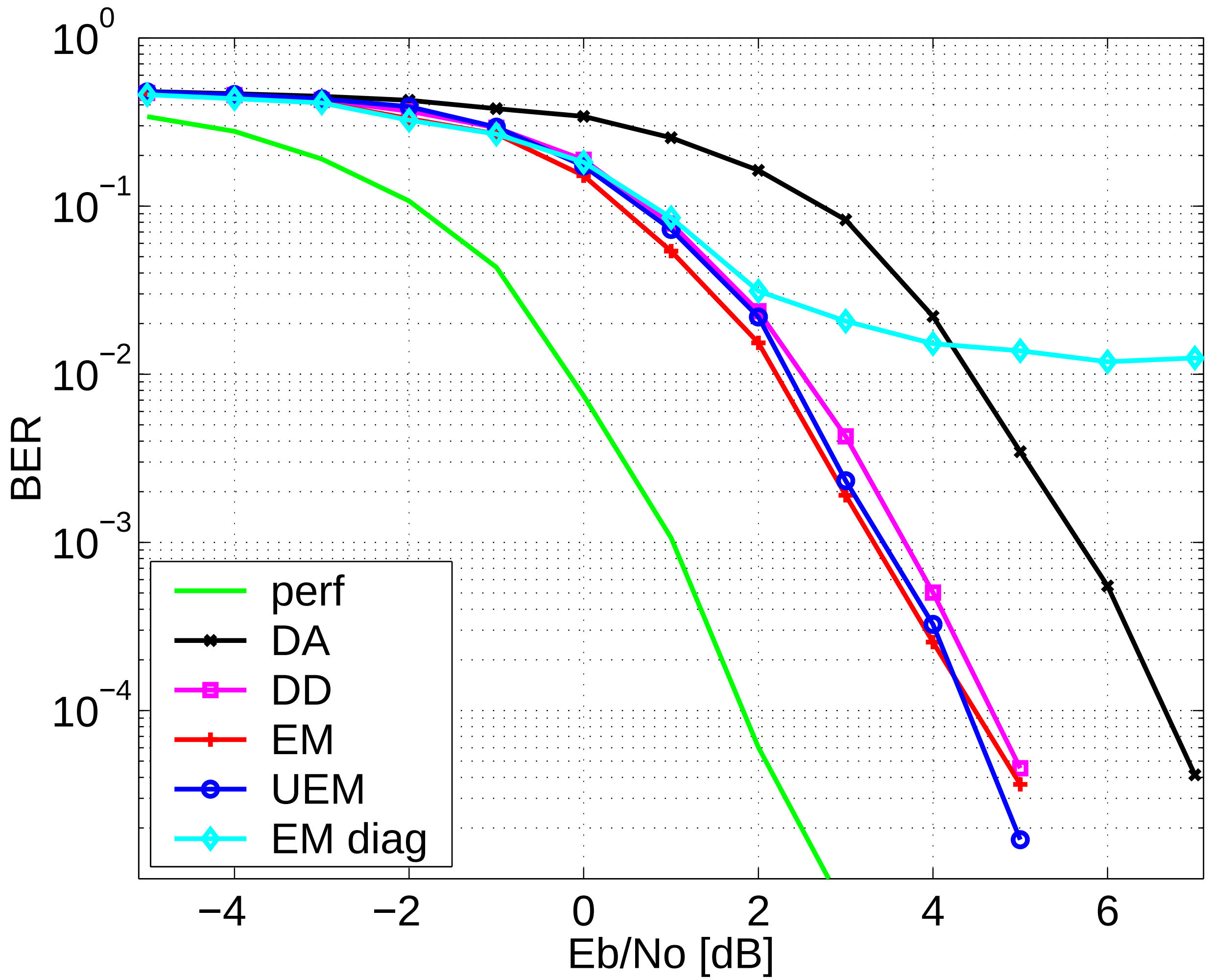
Results for Flat $4 * 4$ MIMO@10 it



Results for Flat $4 * 4$ MIMO



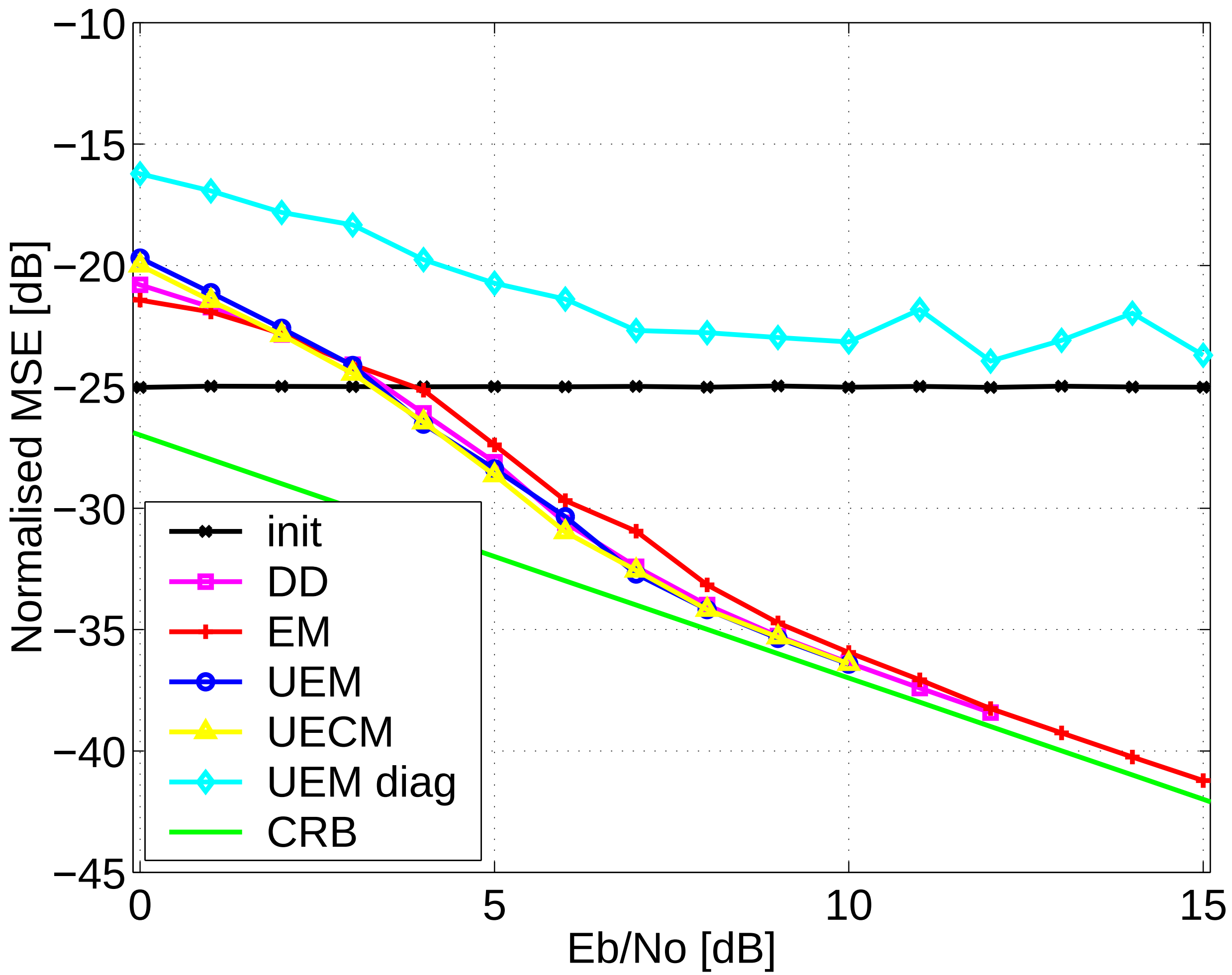
Results for FS $4 * 4$ MIMO



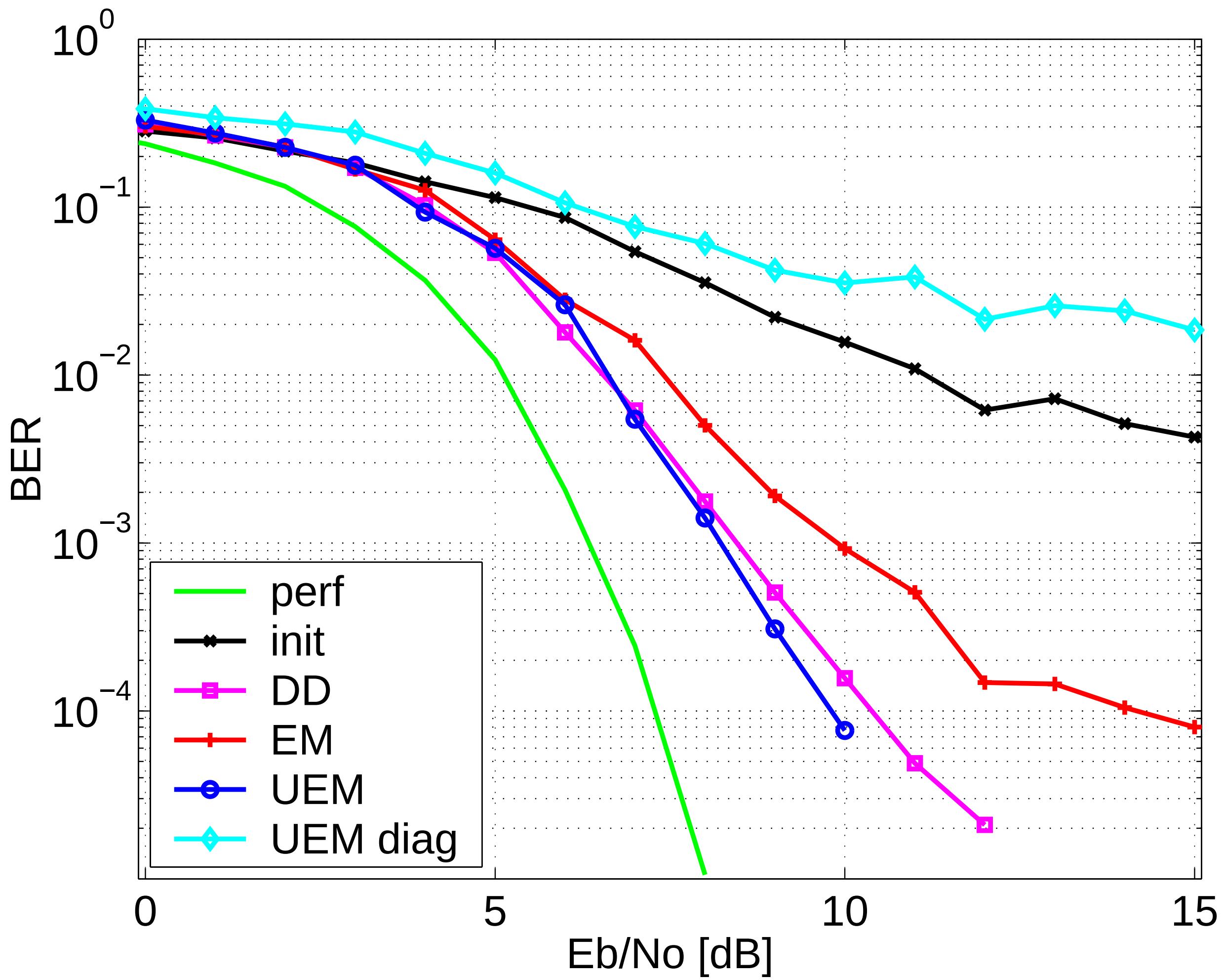
Simulation setup

- Space time BICM, 16-QAM, Gray Mapping
- $r = 0.5$ convolutional encoder, generator polynomials $[7_8, 5_8]$
- frame: 1001 information symbols
- Initialization with CSI corrupted by noise: normalized MSE of $-25dB$
- FS Hiperlan 2/B channel 2×2

Results for Hiperlan II $2 * 2$ channel



Results for Hiperlan II $2 * 2$ channel



Global conclusions

- EM : nice framework for the use of soft information in a synchronization/parameter estimation context
- Improvements have to be introduced wrt pure EM design

Outline

- Introduction/motivation
- The EM algorithm
- Coding and the MAP algorithm
- Synchronization of coded systems with the EM algorithm
- Illustration of performance
- CSI estimation for coded MIMO transmission
- Illustration and performance
- **Cramer-Rao bound with coded/prior information**

Cramer-Rao bound

- Channel with n_T inputs and n_R outputs; bursts of $n_T L_s$ complex symbols $s_k^{(i)}$ are sent
- Model:

$$r_k^{(j)} = \sum_{i=1}^{n_T} \sum_{l=0}^{L-1} h_l^{(i,j)} \, s_{k-l}^{(i)} + n_k^{(j)}, \tag{41}$$

- Let

$$\underline{h}_R = [\Re\{\underline{h}\}^T \, \Im\{\underline{h}\}^T]^T. \tag{42}$$

- We have

$$E_{\underline{r}|\underline{h}}[(\hat{\underline{h}}_R - \underline{h}_R)(\hat{\underline{h}}_R - \underline{h}_R)^T] \geq \mathsf{CRB}(\underline{h}_R). \tag{43}$$

$$\mathsf{CRB}(\underline{h}_R) = \underline{\underline{J}}^{-1}(\underline{h}_R). \tag{44}$$

- Fisher Information Matrix

$$\{\underline{\underline{J}}(\underline{h}_R)\}_{l,k} = E_{\underline{r}|\underline{h}_R}\left[\frac{\partial \ln p(\underline{r}|\tilde{\underline{h}}_R)}{\partial\{\tilde{\underline{h}}_R\}_l} \frac{\partial \ln p(\underline{r}|\tilde{\underline{h}}_R)}{\partial\{\tilde{\underline{h}}_R\}_k}\right]_{|\tilde{\underline{h}}_R = \underline{h}_R}, \tag{45}$$

Cramer-Rao bound

- With nuisance (data) parameters:

$$p(\underline{r}|\underline{\tilde{h}}_R) = \int p(\underline{r}|\underline{\tilde{h}}_R, \underline{s})\, p(\underline{s}) \mathsf{d}\underline{s} \tag{46}$$

- We have

$$\frac{\partial \ln p(\underline{r}|\underline{\tilde{h}}_R)}{\partial \{\underline{\tilde{h}}_R\}_l} = \frac{1}{p(\underline{r}|\underline{\tilde{h}}_R)} \frac{\partial p(\underline{r}|\underline{\tilde{h}}_R)}{\partial \{\underline{\tilde{h}}_R\}_l} \tag{47}$$

- So use the substitution

$$\frac{\partial p(\underline{r}|\underline{\tilde{h}}_R)}{\partial \{\underline{\tilde{h}}_R\}_l} = p(\underline{r}|\underline{\tilde{h}}_R) \frac{\partial \ln p(\underline{r}|\underline{\tilde{h}}_R)}{\partial \{\underline{\tilde{h}}_R\}_l} \tag{48}$$

Cramer-Rao bound

- With nuisance (data) parameters:

$$\frac{\partial \ln p(\underline{r}|\underline{\tilde{h}}_R)}{\partial\{\underline{\tilde{h}}_R\}_l} = \frac{1}{p(\underline{r}|\underline{\tilde{h}}_R)} \frac{\partial p(\underline{r}|\underline{\tilde{h}}_R)}{\partial\{\underline{\tilde{h}}_R\}_l} \tag{49}$$

$$= \frac{1}{p(\underline{r}|\underline{\tilde{h}}_R)} \frac{\partial \int p(\underline{r}|\underline{\tilde{h}}_R, \underline{s})\, p(\underline{s})\mathsf{d}\underline{s}}{\partial\{\underline{\tilde{h}}_R\}_l} \tag{50}$$

$$= \frac{1}{p(\underline{r}|\underline{\tilde{h}}_R)} \int p(\underline{s}) \frac{\partial p(\underline{r}|\underline{\tilde{h}}_R, \underline{s})\, \mathsf{d}\underline{s}}{\partial\{\underline{\tilde{h}}_R\}_l} \tag{51}$$

$$= \int \frac{p(\underline{s})p(\underline{r}|\underline{\tilde{h}}_R, \underline{s})}{p(\underline{r}|\underline{\tilde{h}}_R)} \frac{\partial \ln p(\underline{r}|\underline{\tilde{h}}_R, \underline{s})}{\partial\{\underline{\tilde{h}}_R\}_l} \mathsf{d}\underline{s} \tag{52}$$

$$= \int p(\underline{s}|\underline{\tilde{h}}_R, \underline{r}) \frac{\partial \ln p(\underline{r}|\underline{\tilde{h}}_R, \underline{s})}{\partial\{\underline{\tilde{h}}_R\}_l} \mathsf{d}\underline{s} \tag{53}$$

Cramer-Rao bound

- The effect of the prior distribution of nuisance parameters $\underline{s}$ is captured through the posterior probability
- This posterior probability $p(\underline{s}|\tilde{\underline{h}}_R, \underline{r})$ is exactly what is delivered by an $\tilde{\underline{h}}_R$-aided MAP receiver
- Basic formula for CRB computation over coded system
- Assumes exact posterior probabilities are delivered: true MAP (turbo ?)

Cramer-Rao bound

- About the partial derivatives

$$\frac{\partial \ln p(\underline{r}|\underline{\tilde{h}},\underline{s})}{\partial \Re\{\tilde{h}_l^{(i,j)}\}}_{|\underline{\tilde{h}}=\underline{h}} = \frac{2}{\sigma_n^2}\sum_{k=1}^{L_s}\Re\{s_{k-l}^{(i)*}r_k^{(j)} - \sum_{i'=1}^{n_T}\sum_{l'=0}^{L-1} h_{l'}^{(i',j)} s_{k-l'}^{(i')} s_{k-l}^{(i)*}\}$$

$$\frac{\partial \ln p(\underline{r}|\underline{\tilde{h}},\underline{s})}{\partial \Im\{\tilde{h}_l^{(i,j)}\}}_{|\underline{\tilde{h}}=\underline{h}} = \frac{2}{\sigma_n^2}\sum_{k=1}^{L_s}\Im\{s_{k-l}^{(i)*}r_k^{(j)} - \sum_{i'=1}^{n_T}\sum_{l'=0}^{L-1} h_{l'}^{(i',j)} s_{k-l'}^{(i')} s_{k-l}^{(i)*}\}.$$

- Using $\eta_k^{(i)} = E_{\underline{s}|\underline{r},\underline{h}_R}[s_k^{(i)}]$ and $\rho_{k,k'}^{(i,i')} = E_{\underline{s}|\underline{r},\underline{h}_R}[s_k^{(i)}\ s_{k'}^{(i')*}]$, we finally have

$$\frac{\partial \ln p(\underline{r}|\underline{\tilde{h}})}{\partial \Re\{\tilde{h}_l^{(i,j)}\}}_{|\underline{\tilde{h}}=\underline{h}} = \frac{2}{\sigma_n^2}\sum_{k=1}^{L_s}\Re\{\eta_{k-l}^{(i)*}r_k^{(j)} - \sum_{i'=1}^{n_T}\sum_{l'=0}^{L-1} h_{l'}^{(i',j)} \rho_{k-l',k-l}^{(i',i)}\}$$

$$\frac{\partial \ln p(\underline{r}|\underline{\tilde{h}})}{\partial \Im\{\tilde{h}_l^{(i,j)}\}}_{|\underline{\tilde{h}}=\underline{h}} = \frac{2}{\sigma_n^2}\sum_{k=1}^{L_s}\Im\{\eta_{k-l}^{(i)*}r_k^{(j)} - \sum_{i'=1}^{n_T}\sum_{l'=0}^{L-1} h_{l'}^{(i',j)} \rho_{k-l',k-l}^{(i',i)}\}.$$

Cramer-Rao bound for given mutual information

- Instead of setting $p(\underline{s})$ for each sequence or symbol, one can instead assume a pdf for the symbol probability
- Usually LLR are gaussian distributed
- One can set the mutual information (MI) between $p(\underline{s})$ and the sequence
- Amounts to fixing the LLR distribution : MI=0 $\leftrightarrow$ NDA; MI=1 $\leftrightarrow$ DA
- For a given MI, one has a lower bound given on the CRB given by

$$E_{\underline{r}|\underline{h},\mathrm{MI}}[(\underline{\hat{h}}_R - \underline{h}_R)(\underline{\hat{h}}_R - \underline{h}_R)^T] \geq E_{p(\underline{s})|\mathrm{MI}}[\underline{\underline{J}}^{-1}(\underline{h}_R)], \tag{54}$$

- With Jensen's inequality for matrices:

$$\begin{aligned} E_{\underline{r}|\underline{h},\mathrm{MI}}[(\underline{\hat{h}}_R - \underline{h}_R)(\underline{\hat{h}}_R - \underline{h}_R)^T] &\geq (E_{p(\underline{s})|\mathrm{MI}}[\underline{\underline{J}}(\underline{h})])^{-1} && (55) \\ &= \mathrm{CRB}_{\mathrm{MI}} && (56) \end{aligned}$$

Cramer-Rao bound for random channel

- For an *estimate unbiased on average* : $E_{\underline{h}_R,\underline{r}}[\hat{\underline{h}}_R] = m_{\underline{h}_R}$,
- Lower bound given by

$$E_{\underline{h},\underline{r}}[(\hat{\underline{h}}_R - \underline{h}_R)(\hat{\underline{h}}_R - \underline{h}_R)^T] \geq \mathsf{CRB}_{\mathsf{Rand}}. \tag{57}$$

- with

$$\mathsf{CRB}_{\mathsf{Rand}} = (E_{\underline{h}_R}[\underline{\underline{J_2}}(\underline{h}_R)])^{-1}, \tag{58}$$

- and $\underline{\underline{J_2}}(\underline{h}_R)$ is a matrix whose elements are

$$\{\underline{\underline{J_2}}(\underline{h}_R)\}_{l,k} = E_{\underline{r}|\underline{h}_R}\left[\frac{\partial \ln p(\underline{r}, \tilde{\underline{h}}_R)}{\partial\{\tilde{\underline{h}}_R\}_l} \frac{\partial \ln p(\underline{r}, \tilde{\underline{h}}_R)}{\partial\{\tilde{\underline{h}}_R\}_k}\right]_{|\tilde{\underline{h}}_R=\underline{h}_R}. \tag{59}$$

- Valid for estimators knowing the prior channel distribution or the joint pdf $p(\underline{r}, \tilde{\underline{h}}_R)$

Cramer-Rao bound for random channel

- For a *conditionally unbiased estimator* : $E_{\underline{r}|\underline{h}_R}[\underline{\hat{h}}_R] = \underline{h}_R$.

$$E_{\underline{r},\underline{h}_R}[(\underline{\hat{h}}_R - \underline{h}_R)(\underline{\hat{h}}_R - \underline{h}_R)^T] \geq \text{CRB}_{\text{CU}}, \tag{60}$$

$$\text{CRB}_{\text{CU}} = E_{\underline{h}_R}[\underline{\underline{J}}^{-1}(\underline{h}_R)]. \tag{61}$$

- J of the "usual" CRB (see 44)
- Averaging over r and channel NOT simultaneous (inversion in between)
- With Jensen's inequality for matrices:

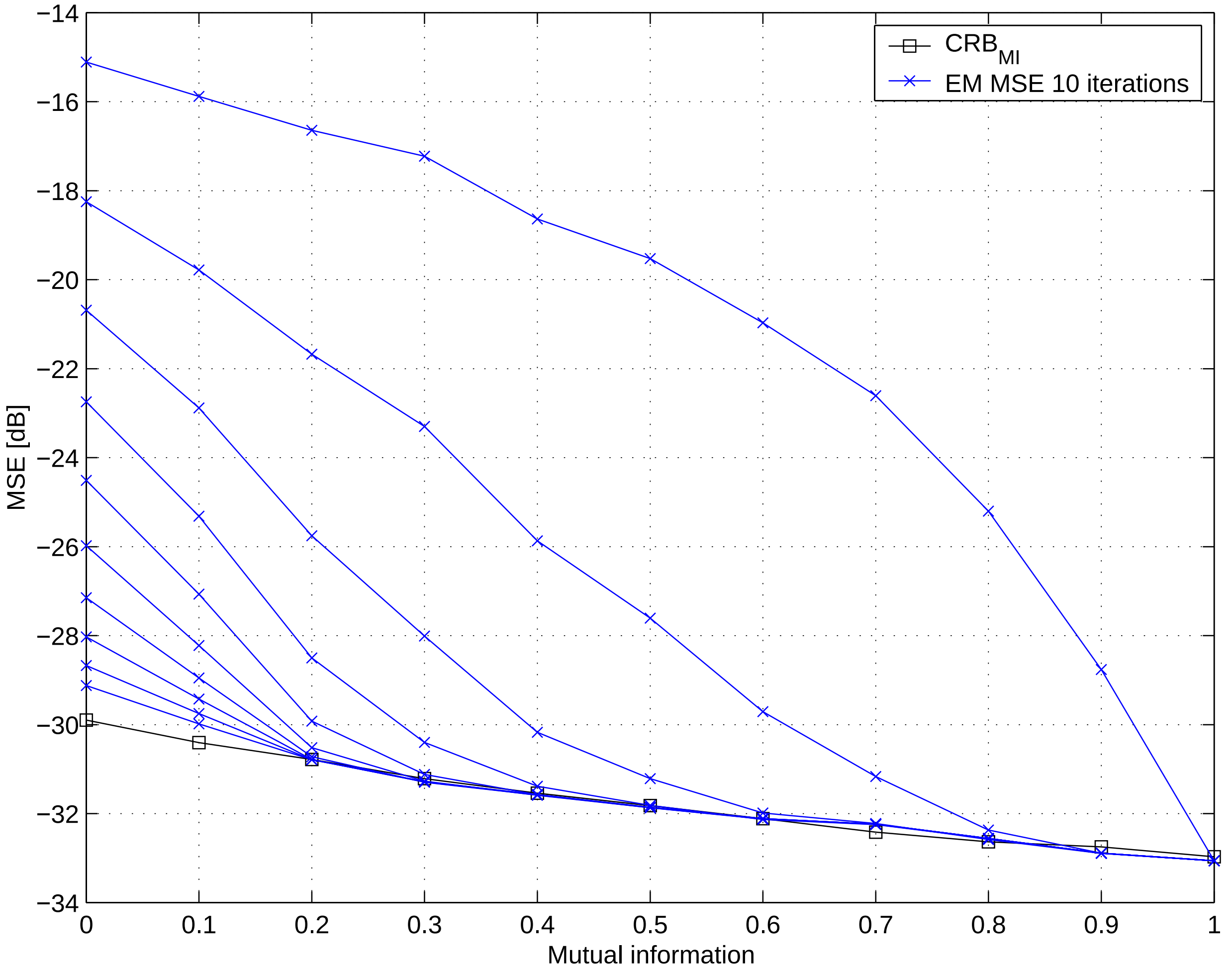
$$\text{CRB}_{\text{CU2}} = (E_{\underline{h}_R}[\underline{\underline{J}}(\underline{h}_R)])^{-1} \tag{62}$$

$$\text{CRB}_{\text{CU2}} \leq \text{CRB}_{\text{CU}}. \tag{63}$$

Results

- Burst sent over Porat channel
- MAP equalizer, no coding, BPSK
- $E_s/N_0 = 0$ dB
- CRB decreases with increasing MI (means closer to DA mode)
- Result also for EM estimation: achieves the CRB after 10 iterations

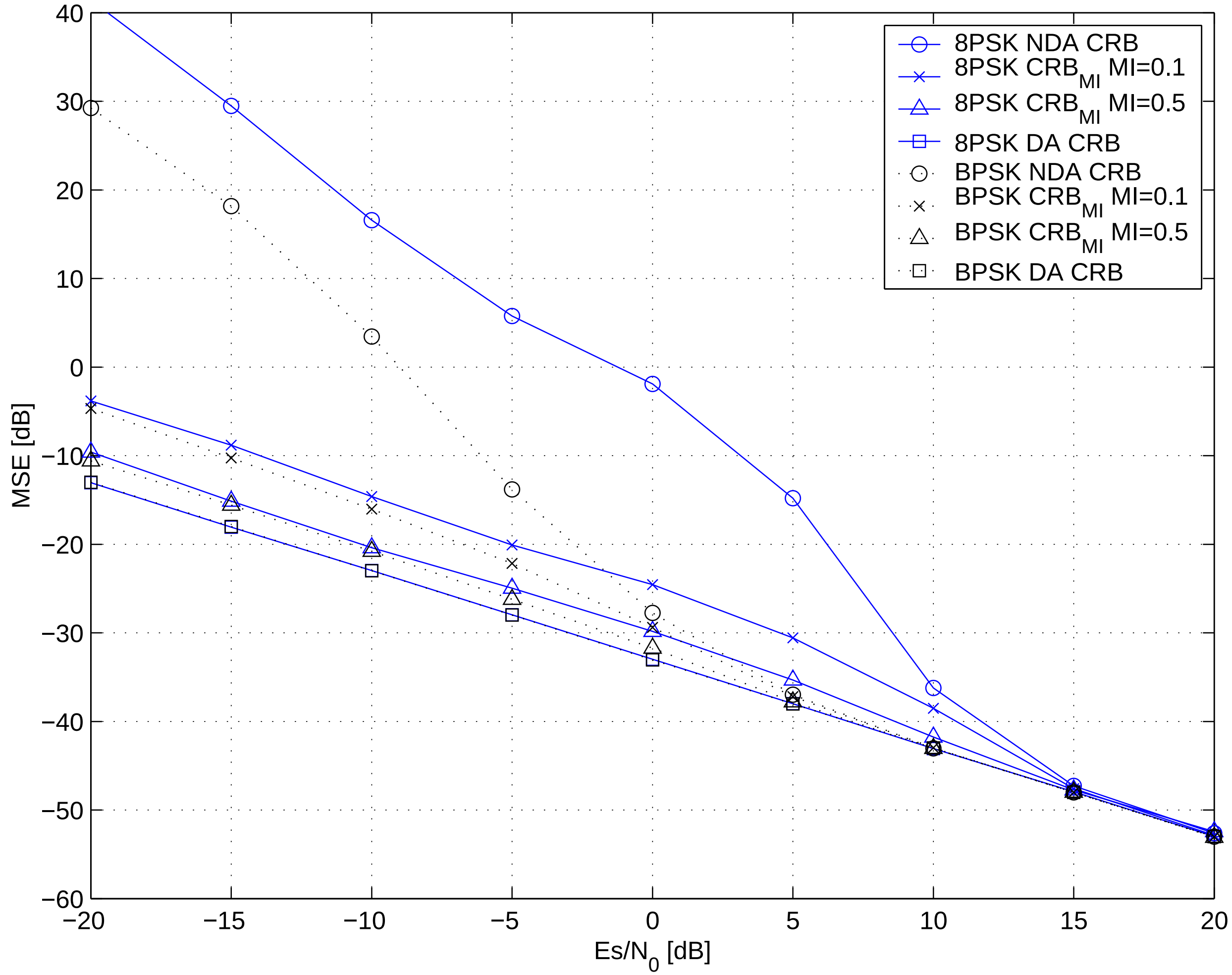
Results for Porat channel



Results for different constellations

- SISO Proakis B channel
- All bounds converge to the DA CRB at high E_s/N_0
- For MI=0.1, smaller constellation better: less uncertainty about symbols for low E_s/N_0
- For large MI information brought by constellation less crucial
- All same DA CRB

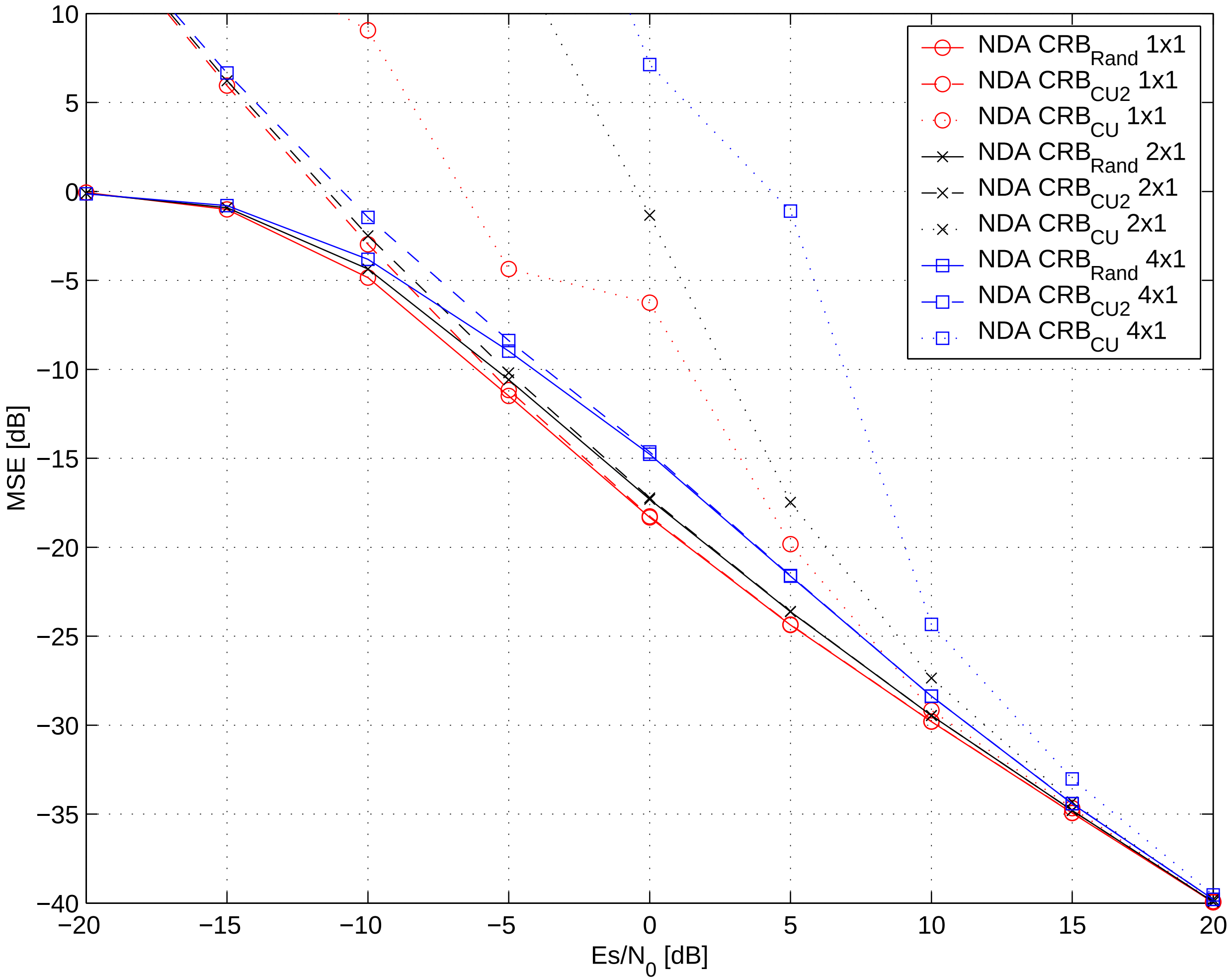
Results for Proakis B



Results for random channels

- Flat Rayleigh fading
- 1,2 or 4 TX antennas
- All NDA: MI=0
- Benefitial knowledge of channel distribution for low E_s/N_0
- Degradation with increasing number of antennas: less information about data (more interference)

Results for MISO flat Rayleigh



Thank you !

Parameter estimation for the Alamouti scheme: impact of diversity on ”estimability”

L. Vandendorpe (UCL)

Thanks to J. Louveaux

UCL
Université catholique de Louvain

Outline

- Introduction/motivation
- Alamouti scheme
- CRB and nuisance parameters
- Results for Alamouti

Outline

- **Introduction/motivation**
- Alamouti scheme
- CRB and nuisance parameters
- Results for Alamouti

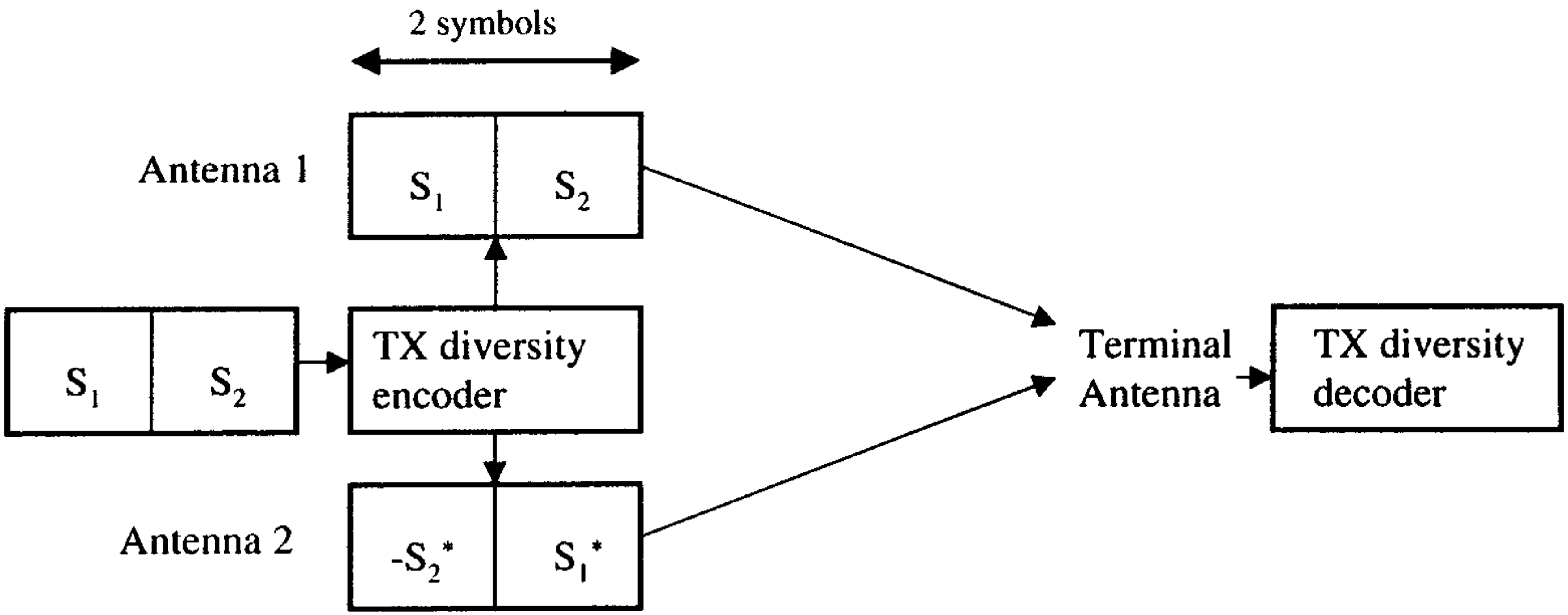
Motivation

- Alamouti benefits from order 2 diversity
- Effect known for detection: slope of BER curve changes accordingly
- What about sensitivity to synchronisation errors ?
- **Does diversity impact the sensitivity and the CRB ?**

Outline

- Introduction/motivation
- **Alamouti scheme**
- CRB and nuisance parameters
- Results for Alamouti

Transmitter



Model

- Transmitted signal (baseband)

$$x_0(t) = \sum_{n=0}^{N-1} [s_0(n)u(t-2nT) - s_1^*(n)u(t-2nT-T)] \quad (1)$$

$$x_1(t) = \sum_{n=0}^{N-1} [s_1(n)u(t-2nT) + s_0^*(n)u(t-2nT-T)] \quad (2)$$

- Received signal

$$\begin{aligned} r(t) &= h_0 \sum_{n=0}^{N-1} [s_0(n)u(t-2nT-\tau) - s_1^*(n)u(t-2nT-T-\tau)] \\ &+ h_1 \sum_{n=0}^{N-1} [s_1(n)u(t-2nT-\tau) + s_0^*(n)u(t-2nT-T-\tau)] \\ &+ n(t) \end{aligned} \quad (3)$$

Question

- h_0, h_1 are both complex circular gaussian (Rayleigh fading)
- What is the impact on the "estimability" of τ
- To be compared with a non diversity situation

Transmitter

- Transmitted signal

$$x(t) \;=\; \sum_{n=0}^{N-1} s(n)u(t-nT-\tau) \tag{4}$$

- Received signal

$$r(t) \;=\; h\sum_{n=0}^{N-1} s(n)u(t-nT-\tau)+n(t) \tag{5}$$

- with

$$s(n) \;=\; s_r(n)+js_i(n) \tag{6}$$

Likelihood function

- Assuming h

$$p[r;\tau|h_r,h_i] = C\ \exp[\int_{-\infty}^{\infty} -|r(t)-h\sum_{n=0}^{N-1} s(n)u(t-nT-\tau)|^2/2N_0] \quad (7)$$

- After expansion/simplification

$$p'[r;\tau|h_r,h_i] = C\ \exp[h_r\,A_r/N_0+h_i\,A_i/N_0]\ \exp[-(h_r^2+h_i^2)B/2N_0] \quad (8)$$

$$A_r = \sum_n [s_r(n)y_r(n) + s_i(n)y_i(n)] \quad (9)$$

$$A_i = \sum_n [s_r(n)y_i(n) - s_i(n)y_r(n)] \quad (10)$$

$$B = \sum_n |s(n)|^2 \quad (11)$$

$$y(n) = y_r(n) + jy_i(n) = \int_{-\infty}^{\infty} r(t)u^*(t-nT-\tau)\,\mathrm{d}t \quad (12)$$

Outline

- Introduction/motivation
- Alamouti scheme
- **CRB and nuisance parameters**
- Results for Alamouti

Cramér Rao bound

- The CRB: (for any unbiased estimator):

$$\sigma_{\hat{\tau}}^2 \geq \frac{1}{-\mathbf{E}\left[\frac{\partial^2 \ln p[r;\tau]}{\partial \tau^2}\right]} \tag{13}$$

- where $\mathbf{E}[.]$ means is expectation wrt to $p[r;\tau]$
- How to handle h, or a nuisance parameter ?
- 4 possible cases

CRB for case 1: joint estimation

- If nothing is known about h, should be estimated together with τ
- Compute the Fisher information matrix J with ($\underline{\theta}^T = [\tau, h_r, h_i]$)

$$J_{i,j} = \mathbf{E}\left[\frac{\partial \ln p[r;\underline{\theta}]}{\partial \theta_i}\frac{\partial \ln p[r;\underline{\theta}]}{\partial \theta_j}\right] = -\mathbf{E}\left[\frac{\partial^2 \ln p[r;\underline{\theta}]}{\partial \theta_i \partial \theta_j}\right] \tag{14}$$

$$\sigma^2_{\hat{\theta}_i} \geq \left[J^{-1}\right]_{ii} \tag{15}$$

- where $\mathbf{E}[.]$ means is expectation wrt to $p[r;\underline{\theta}]$
- Not interesting here: we want the effect of the distribution of h

CRB for case 2: nuisance parameter

- h has to be "removed" in the likelihood function
- Situation comparable with the symbols
- Called "nuisance parameters"
- "Proper" handling of nuisance
- Averaging over h

$$
\begin{aligned}
p[r;\tau] &= \int_{h_r} \mathrm{d}h_r \int_{h_i} \mathrm{d}h_i \, T_{h_r,h_i}(h_r,h_i)\, p[r;\tau|h_r,h_i] && (16)\\
&= C' \exp[\alpha^2 \sum_n |s^*(n)y(n)|^2] && (17)
\end{aligned}
$$

$$
\alpha^2 = \frac{1}{2N_0^2}\left[\frac{1}{\sigma_h^2} + \frac{\sum_n |s(n)|^2}{N_0}\right]^{-1} \tag{18}
$$

CRB for case 2: nuisance parameter (cont'd)

- This corresponds to a "non-h-aided solution"; for any estimator that does not use the knowledge (estimation) of h
- The CRB: (for any unbiased estimator):

$$\sigma^2_{\hat{\tau}} \geq \frac{1}{-\mathbf{E}\left[\frac{\partial^2 \ln p[r;\tau]}{\partial \tau^2}\right]} \tag{19}$$

- where $\mathbf{E}[.]$ means is expectation wrt to $p[r;\tau]$

CRB for case 3: h aided solution

- Assume h is known and compute the h-aided CRB for τ

$$\sigma^2_{h,\hat{\tau}} \geq \frac{1}{-\mathbf{E}\left[\dfrac{\partial^2 \ln p[r;\tau,h_r,h_i]}{\partial \tau^2}\right]} \qquad (20)$$

- Then compute the average of this CRB over the statistics of h

$$\sigma^2_{MCB,\hat{\tau}} = \int_{h_r} \mathrm{d}h_r \int_{h_i} \mathrm{d}h_i \, T_{h_r,h_i}(h_r,h_i) \frac{1}{-\mathbf{E}\left[\dfrac{\partial^2 \ln p[r;\tau,h_r,h_i]}{\partial \tau^2}\right]} \qquad (21)$$

CRB for case 4: bound modified wrt h

- Compute $p[r;\tau,h_r,h_i]$
- Compute

$$\sigma^2_{m,\hat{\tau}} \geq \frac{1}{-\mathbf{E}_{r,h_r,h_i}\left[\frac{\partial^2 \ln p[r;\tau,h_r,h_i]}{\partial \tau^2}\right]} \tag{22}$$

- where $\mathbf{E}_{r,h_r,h_i}[.]$ means expectation wrt to both r and h

Outline

- Introduction/motivation
- Alamouti scheme
- CRB and nuisance parameters
- **Results for Alamouti**

Discussion

- Cases 2 and 4: same solution for Alamouti or non Alamouti !
- If normalization such that identical number of symbols, and total emitted power
- Value for MCRB:

$$\left(\frac{E_s}{N_0}\right)^{-1} \frac{1}{N_{na} W_s^2} \tag{23}$$

$$\bar{E}_s = 2\sigma_h^2 \sigma_s^2 \frac{1}{2\pi} \int_{-\infty}^{\infty} \mathrm{d}\omega \, |U(\omega)|^2 \tag{24}$$

$$W_s^2 = \frac{\int_{-\infty}^{\infty} \mathrm{d}\omega \, \omega^2 \, |U(\omega)|^2}{\int_{-\infty}^{\infty} \mathrm{d}\omega \, |U(\omega)|^2} \tag{25}$$

Discussion

- Apparently: no benefit from diversity when non h aided solution
- Is this logical ? Yes
- One should remember that the detector providing diversity IS h aided
- A non-h aided detector would maximize (see above)

$$p[r;\tau] = C' \exp[\alpha^2 \sum_n |s^*(n)y(n)|^2] \tag{26}$$

- Something similar for non h aided Alamouti detection
- So the diversity in detection is measured by considering the h aided detector and then average the $BER(h)$ over the statistics of h
- One should "mimic" this for estimation

h-aided Alamouti detector

- Detection structure:

$$\hat{s}_0(n') = h_0^* \int_{-\infty}^{\infty} r(t)u(t-2n'T)\mathrm{d}t + h_1^* \left[\int_{-\infty}^{\infty} r(t)u(t-2n'T-T)\mathrm{d}t\right]^* \quad (27)$$

$$\hat{s}_1(n') = h_1^* \int_{-\infty}^{\infty} r(t)u(t-2n'T)\mathrm{d}t - h_0 \left[\int_{-\infty}^{\infty} r(t)u(t-2n'T-T)\mathrm{d}t\right]^* \quad (28)$$

- Structure of decision variables

$$\hat{s}_0(n') = \left[|h_0|^2 + |h_1|^2\right] s_0(n') + h_0^*\nu_0(n) + h_1\nu_1^*(n) \quad (29)$$

$$\hat{s}_1(n') = \left[|h_0|^2 + |h_1|^2\right] s_1(n') + h_1^*\nu_0(n) - h_0\nu_1^*(n) \quad (30)$$

Impact of diversity on error bound

- For Q-QAM modulation, symbol error bounded by :

$$P_b < 2\,(1 - \frac{1}{\sqrt{Q}})\,\exp^{-\frac{3\,SNR}{2(Q-1)}} \tag{31}$$

- Averaging over the SNR distribution normalized such that the average received energy is constant, it comes for Alamouti

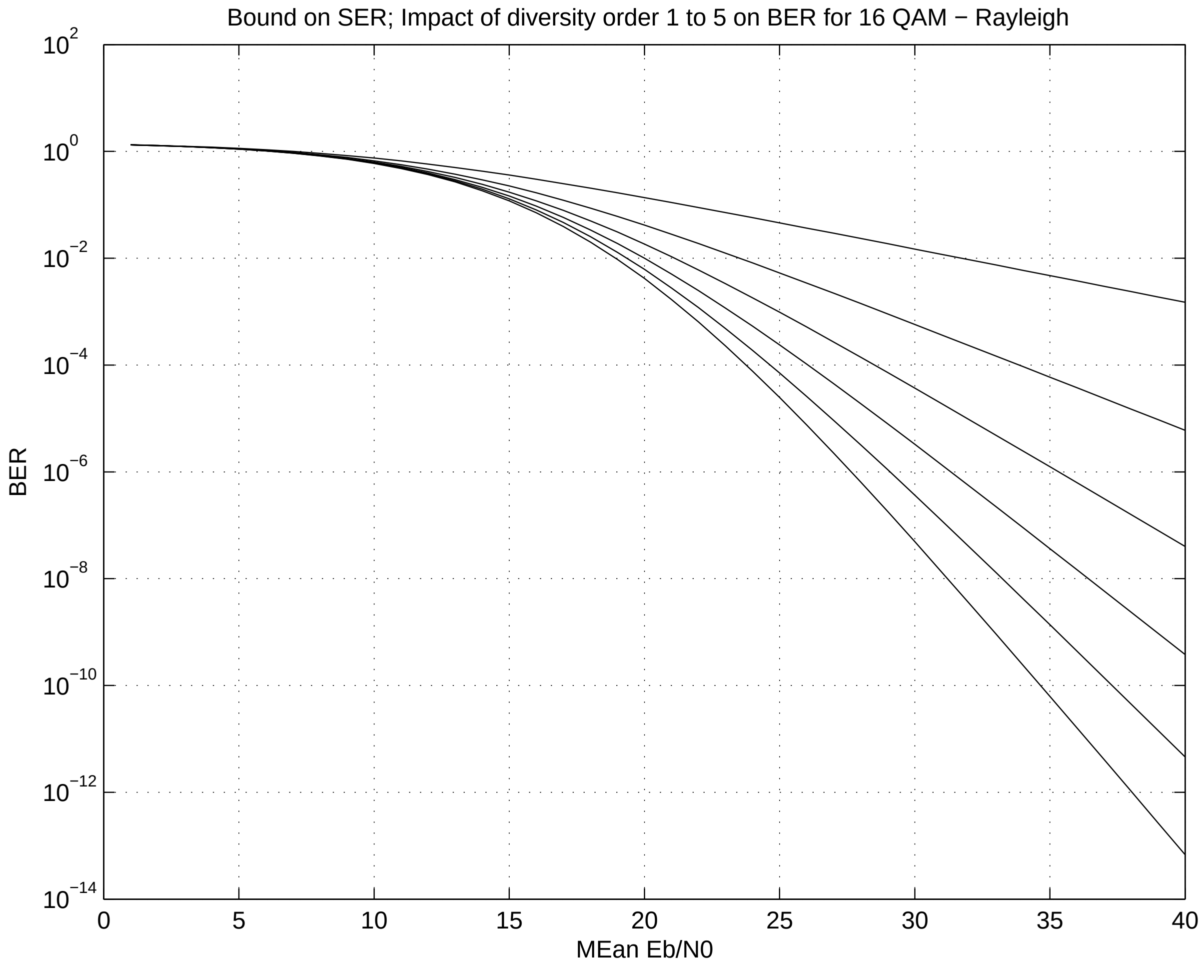
$$\bar{P}_b < 2\,(1 - \frac{1}{\sqrt{Q}})\left[\frac{0.75}{Q-1}\frac{\bar{E}_s}{N_0} + 1\right]^{-2} \tag{32}$$

- For non Alamouti

$$\bar{P}_b < 2\,(1 - \frac{1}{\sqrt{Q}})\left[\frac{1.5}{Q-1}\frac{\bar{E}_s}{N_0} + 1\right]^{-1} \tag{33}$$

- where $\bar{E}_s$ is the average received energy per branch in the non-Alamouti case
- slope of the SER determined by diversity order: this is how diversity materializes !

Illustration for $Q = 16$-QAM and Rayleigh channels



Case 3 Non Alamouti

- Bound for given h_0:

$$\left(\frac{E_s}{N_0}\right)^{-1} \frac{1}{N_{na} W_s^2 |h_0|^2 / 2\sigma_h^2} \tag{34}$$

- $|h_0|^2$ is χ^2 with 2 degrees of freedom
- for $u = |h_0|^2/2\sigma_h^2$,

$$T(u) = \exp^{-u} \text{ and } \int_0^\infty u^{-1} \exp^{-u} \mathrm{d}u = \infty \tag{35}$$

- Average of h-aided bound is infinite

Case 3 Alamouti

- Bound for given h_0, h_1:

$$\left(\frac{E_s}{N_0}\right)^{-1} \frac{4}{N_{na}\, W_s^2\, (|h_0|^2 + |h_1|^2)/\sigma_h^2} \tag{36}$$

- $|h_0|^2 + |h_1|^2$ is χ^2 with 4 degrees of freedom
- for $u = (|h_0|^2 + |h_1|^2)/\sigma_h^2$,

$$T(u) = 0.25\, u \, \exp^{-u/2} \text{ and } \int_0^\infty u^{-1}\, 0.25\, u \, \exp^{-u/2} \mathrm{d}u = 0.5 \tag{37}$$

- Average of h-aided bound is finite and given by

$$\left(\frac{E_s}{N_0}\right)^{-1} \frac{2}{N_{na}\, W_s^2} \tag{38}$$

Thank you !