Lecture 6

Channel Coding

and

Interleaving

1

LORA: Future by Lund

- www.futurebylund.se
- ”The network will be free for those who want to try their products, services and solutions in a pre-commercial stage. This implies that it is only for testing and not for commercial use.”

Channel coding

- Overview
- Block codes
- Convolution codes
- Fading channel and interleaving

Coding is a much more complicated topic than this. Anyone interested should follow a course on channel coding.

LUND UNIVERSITY

3

Simple example

- Data to be sent 1 1 0 1 0 1
- Code: [1] => [1 1 1], [0] => [0 0 0]
- Bit sequency transmitted: 1 1 1 | 1 1 1 | 0 0 0 | 1 1 1 | 0 0 0 | 1 1 1

LUND UNIVERSITY

Simple example

- Data to be sent 1 1 0 1 0 1
- Code: [1] => [1 1 1], [0] => [0 0 0]
- Bit sequency transmitted: 1 1 1 | 1 1 1 | 0 0 0 | 1 1 1 | 0 0 0 | 1 1 1
- Add noise: 0 0 0 | 1 0 0 | 1 0 1 | 1 0 0 | 0 1 0 | 1 0 0

LUND UNIVERSITY

Simple example

- Data to be sent 1 1 0 1 0 1
- Code: [1] => [1 1 1], [0] => [0 0 0]
- Bit sequency transmitted: 1 1 1 | 1 1 1 | 0 0 0 | 1 1 1 | 0 0 0 | 1 1 1
- Add noise: 0 0 0 | 1 0 0 | 1 0 1 | 1 0 0 | 0 1 0 | 1 0 0
- Data recieved: 1 1 1 | 0 1 1 | 1 0 1 | 0 1 1 | 0 1 0 | 0 1 1

LUND UNIVERSITY

Simple example

- Data to be sent 1 1 0 1 0 1
- Code: [1] => [1 1 1], [0] => [0 0 0]

Bit sequency transmitted:	1 1 1	1 1 1	0 0 0	1 1 1	0 0 0	1 1 1
Add noise:	0 0 0	1 0 0	1 0 1	1 0 0	0 1 0	1 0 0
Data recieved:	1 1 1	0 1 1	1 0 1	0 1 1	0 1 0	0 1 1
Decoder (simple):	1	x	x	x	x	x

LUND UNIVERSITY

Simple example

- Data to be sent 1 1 0 1 0 1
- Code: [1] => [1 1 1], [0] => [0 0 0]

Bit sequency transmitted:	1 1 1	1 1 1	0 0 0	1 1 1	0 0 0	1 1 1
Add noise:	0 0 0	1 0 0	1 0 1	1 0 0	0 1 0	1 0 0
Data recieved:	1 1 1	0 1 1	1 0 1	0 1 1	0 1 0	0 1 1
Decoder (simple):	1	x	x	x	x	x
Better decoder (majority).	1	1	1	1	0	1

LUND UNIVERSITY

Channel coding: Basic types of codes

We can classify channel codes in two principal groups:

BLOCK CODES

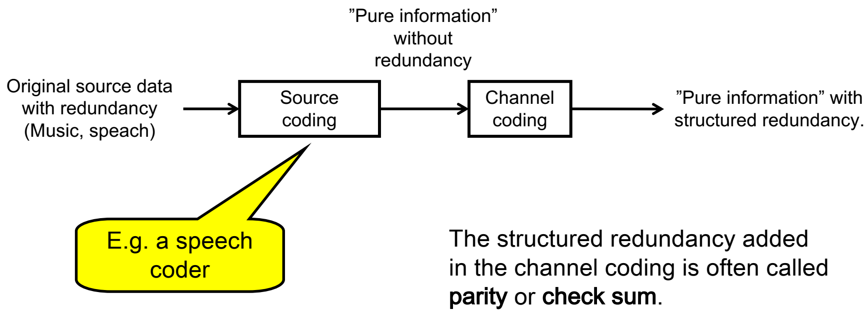
Encodes data in blocks of *k*, using code words of length *n*.

CONVOLUTION CODES

Encodes data in a stream, without breaking it into blocks, creating a code sequence.

LUND UNIVERSITY

9

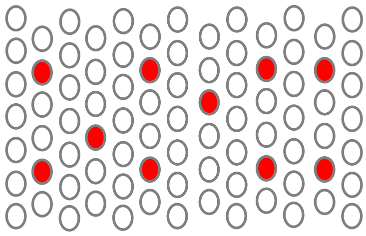


The structured redundancy added in the channel coding is often called **parity** or **check sum**.

LUND UNIVERSITY

10

Assume that we have a block code, which consists of k information bits per n bit code word ($n > k$). Since there are only 2^k different information sequences, there can be only 2^k different code words.

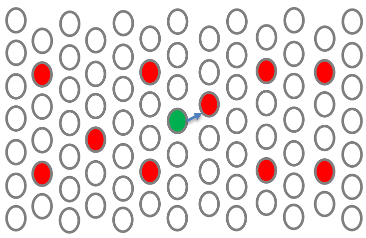


This leads to a larger **distance** between the valid code words than between arbitrary binary sequences of length n, which increases our chance of selecting the correct one after receiving a noisy version.

LUND UNIVERSITY

11

If we receive a sequence that is not a valid code word, we decode to the **closest** one.



One thing remains ... what do we mean by **closest**?
We need a distance measure.

LUND UNIVERSITY

12

Channel Coding Distances

The distance measure used depends on the channel over which we transmit our code words (if we want the rule of decoding to the *closest* code word to give a low probability of error).

Two common ones:

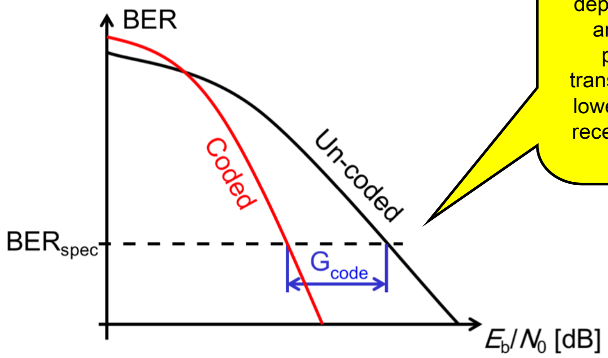
Hamming distance	Measures the number of bits being different between two binary words.	Used for binary channels with random bit errors.
Euclidean distance	Same measure we have used for signal constellations.	Used for AWGN channels.

LUND UNIVERSITY

13

Coding gain

When applying channel codes we decrease the E_b/N_0 required to obtain some specified performance (BER).



This **coding gain** depends on the code and the specified performance. It translates directly to a lower requirement on received power in the link budget.

NOTE: E_b denotes **energy per information bit**, even for the coded case.

LUND UNIVERSITY

14

Channel Coding Bandwidth

When introducing coding we have essentially two ways of handling the indreased number of (code) bits that need to be transmitted:

1) Accept that the raw bit rate will increase the required radio bandwidth proportionally.

This is the simplest way, but may not be possible, since we may have a limited bandwidth available.

2) Increase the signal constellation size to compensate for the increased number of bits, thus keeping the same bandwidth.

Increasing the number of signal constellation points will decrease the distance between them. This decrease in distance will have to be compensated by the introduced coding.

LUND UNIVERSITY

15

Channel coding: Linear block codes

The encoding process of a linear block code can be written as

$$\boldsymbol{x} = \boldsymbol{G}\boldsymbol{u}$$

where

$\boldsymbol{u}$ k - dimensional information vector

$\boldsymbol{G}$ n x k - dimensional generator matrix

$\boldsymbol{x}$ n - dimensional code word vector

The matrix calculations are done in an appropriate arithmetic. We will primarily assume **binary codes** and **modulo-2** arithmetic.

LUND UNIVERSITY

16

Channel coding: Some definitions

Code rate:

$$R = \frac{\text{bits in}}{\text{bits out}} = \frac{k}{n}$$

Modulo-2 arithmetic (XOR):

$$\boldsymbol{x}_i + \boldsymbol{x}_j = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

Hamming weight:

$$w(\boldsymbol{x}) = \text{number of ones in } \boldsymbol{x}$$

Hamming distance:

$$d(\boldsymbol{x}_i, \boldsymbol{x}_j) = w(\boldsymbol{x}_i + \boldsymbol{x}_j)$$

Minimum distance of code:

$$d_{\min} = \min_{i \neq j} d(\boldsymbol{x}_i, \boldsymbol{x}_j) = \min_{i \neq j} w(\boldsymbol{x}_i + \boldsymbol{x}_j)$$

The minimum distance of a code determines its error correcting performance in non-fading channels.

Note: The textbook sometimes use the name "**Hamming distance of the code**" (d_H) to denote its minimum distance.

Encoding example

For a specific (n,k) = (7,4) code we encode the information sequence 1 0 1 1 as

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 1 \\ 1 \\ 0 \\ 1 \\ 1 \end{bmatrix}$$

If the information is directly visible in the code word, we say that the code is **systematic**.

In addition to the k information bits, there are n-k = 3 **parity** bits.

18

Encoding example, cont.

Encoding all possible 4 bit information sequences gives:

Information				Code word							Hamming weight
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1	1	1	1	4
0	0	1	0	0	0	1	0	0	1	1	3
0	0	1	1	0	0	1	1	1	0	0	3
0	1	0	0	0	1	0	0	1	0	1	3
0	1	0	1	0	1	0	1	0	1	0	3
0	1	1	0	0	1	1	0	1	1	0	4
0	1	1	1	0	1	1	1	0	0	1	4
1	0	0	0	1	0	0	0	1	1	0	3
1	0	0	1	1	0	0	1	0	0	1	3
1	0	1	0	1	0	1	0	1	0	1	4
1	0	1	1	1	0	1	1	0	1	0	4
1	1	0	0	1	1	0	0	0	1	1	4
1	1	0	1	1	1	0	1	1	0	0	4
1	1	1	0	1	1	1	0	0	0	0	3
1	1	1	1	1	1	1	1	1	1	1	7

This code has a minimum distance of 3.
(Minimum code word weight of a linear code, excluding the all-zero code word.)

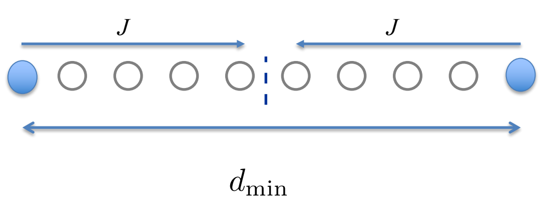
This is a (7,4) Hamming code, capable of correcting one bit error.

Error correction capability

A binary block code with minimum distance d_{min} can correct J bit errors, where

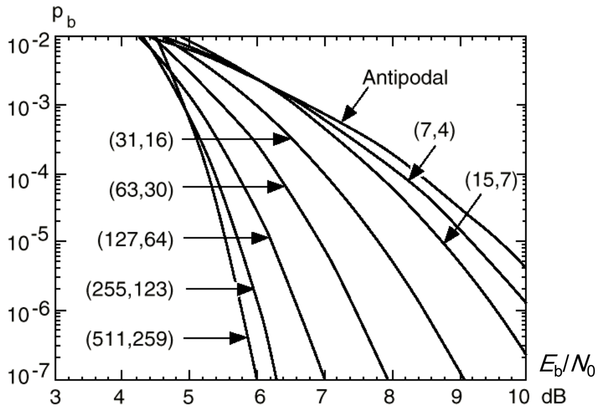
$$J = \left\lfloor \frac{d_{\min} - 1}{2} \right\rfloor$$

Rounded down to nearest integer.



Performance and code length

Longer codes (with same rate) usually have better performance!



Rate ≈ 0.5

This example is for a non-fading channel.

Drawbacks with long codes is complexity and delay.

21

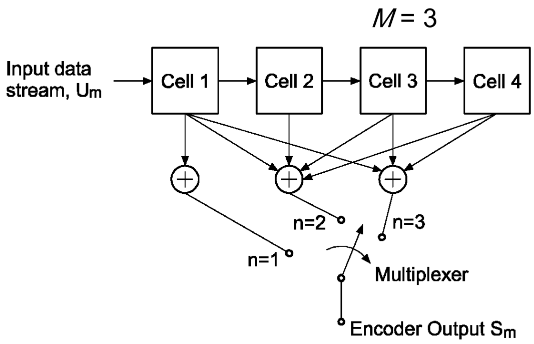
CONVOLUTION CODES

22

Encoder structure

In convolution codes, the coded bits are formed as convolutions between the incoming bits and a number of generator sequences.

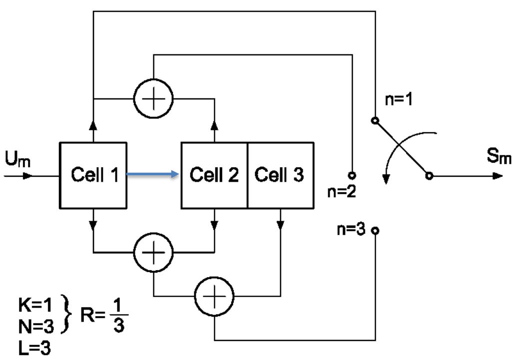
We will view the encoder as a shift register with memory *M,* length *L* and *N* generator sequences (convolution sums).



LUND UNIVERSITY

23

Encoding example



Input	State	Output	Next state
0	00	000	00
1	00	111	10
0	01	001	00
1	01	110	10
0	10	011	01
1	10	100	11
0	11	010	01
1	11	101	11

We usually start the encoder in the **all-zero** state!

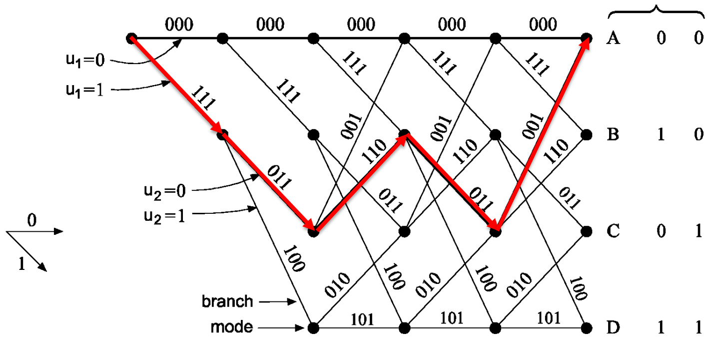
LUND UNIVERSITY

24

Encoding example, cont.

We can view the encoding process in a trellis created from the table on the previous slide.

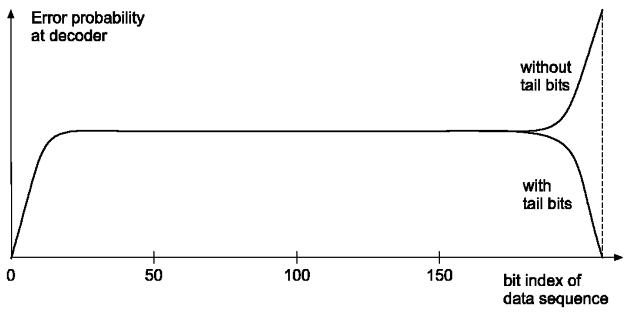
Input (information) bit sequence	1	0	1	0	0
Output (coded) bit sequence	111	011	110	011	001



Termination

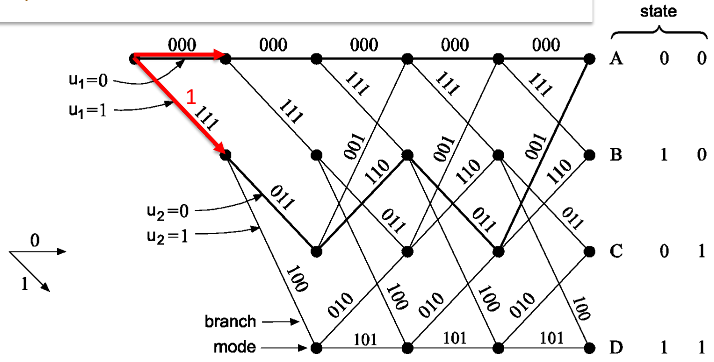
At the end of the information sequence, it is common to add a **tail** of L zeros to force the encoder to end (terminate) in the zero state.

This improved performance, since a decoder knows both the starting state and ending state.



Encoding example, cont.

Recieved sequence: 010 000 1000 001 011 110 001



27

Improvements to Viterbi decoding: Soft decoding

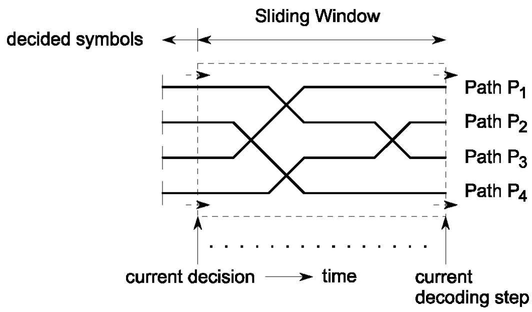
We have given examples of hard decoding, using the **Hamming distance**.

If we do not detect ones and zeros before decoding our channel code, we can use **soft decoding**. In the AWGN channel, this means comparing Euclidean distances instead.

Surviving paths

The Viterbi algorithm needs to keep track of one surviving path per state in the trellis. For long code sequences this causes a memory problem.

In practice we only keep track of surviving paths in a window consisting of a certain number of trellis steps. At the end of this window we enforce decisions on bits, based on the metric in the latest decoding step.



LUND UNIVERSITY

FADING CHANNELS AND INTERLEAVING

LUND UNIVERSITY

30

Fading channels and interleaving

In fading channels, many received bits will be of "low quality" when we hit a fading dip.

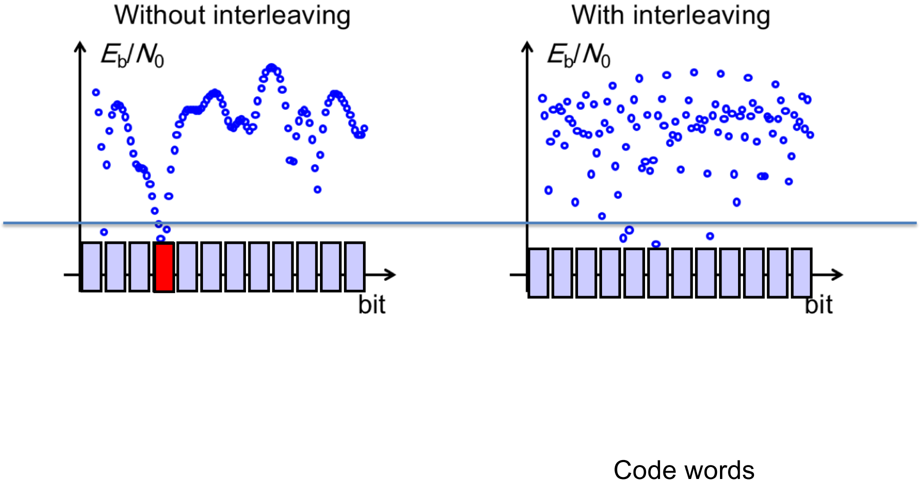
Coding may suffer greatly, since many "low quality" bits in a code word may lead to a decoding error.

To prevent all "low quality" bits in a fading dip from ending up in the same code word, we rearrange the bits between several code words before transmission ... and rearrange them again at the receiver, before decoding.

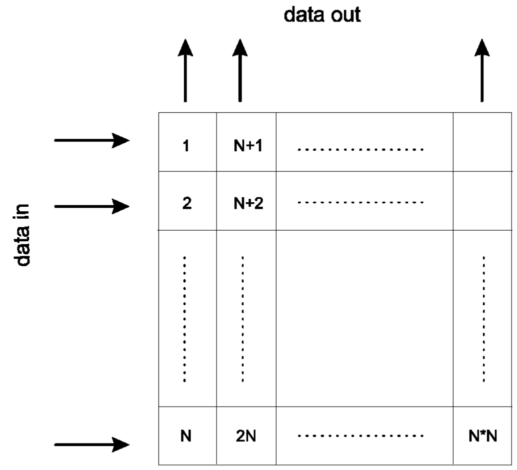
This strategy of breaking up fading dips is called **interleaving**.

31

Distribution of low-quality bits



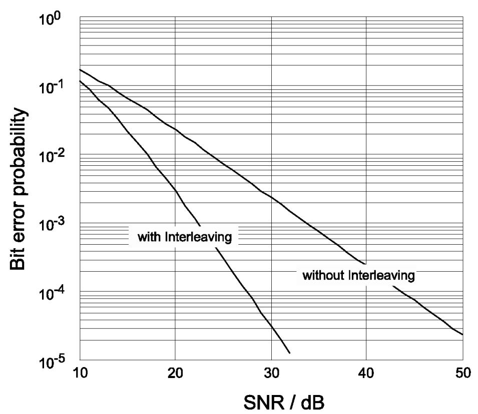
Block interleaver



The writing and reading of data in interleavers cause a delay in the system, which may cause other problems.

Interleaving - BER example

BER of a R=1/3 repetition code over a Rayleigh-fading channel, with and without interleaving. Decoding strategy: majority selection.



LUND
UNIVERSITY