

# Laboratory session 1

## Implementation of random binning encoding and secrecy rate evaluation

Nicola Laurenti, Laura Crosara

April 20, 2023



Except where otherwise stated, this work is licensed under the Creative Commons Attribution-ShareAlike 4.0 International License.

# Laboratory session 1— Contents

Review of random binning encoding

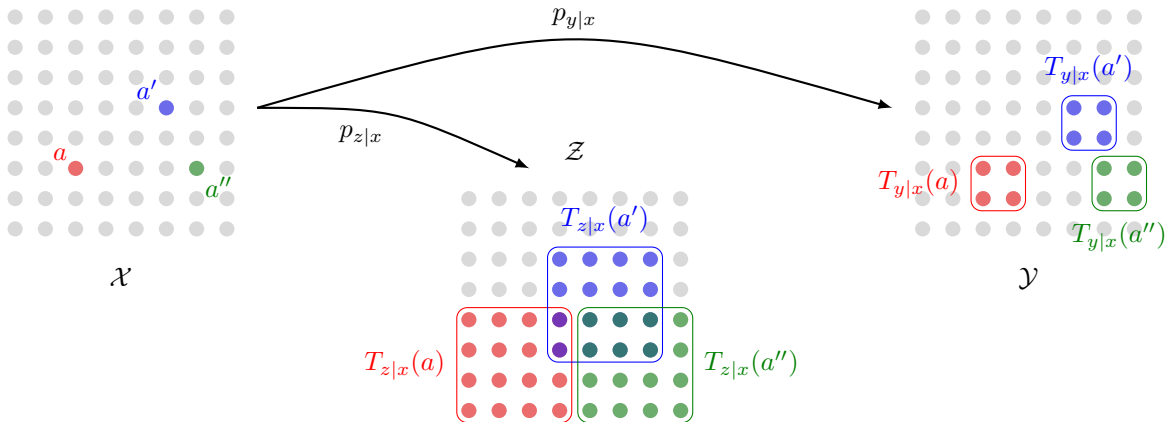
Your tasks in this laboratory session

Appendices

# Toy example: uniform channel

Consider a wiretap channel in which

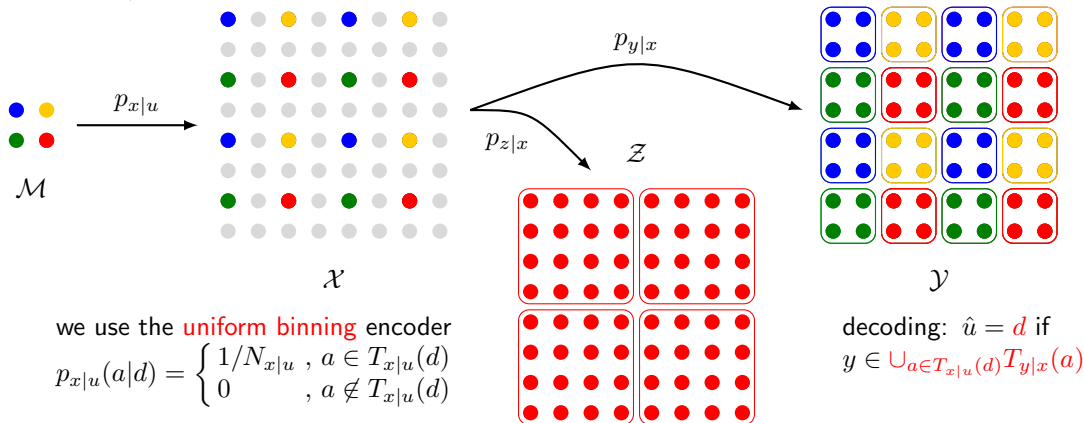
$$p_{y|x}(b|a) = \begin{cases} 1/N_{y|x} & , b \in T_{y|x}(a) \\ 0 & , b \notin T_{y|x}(a) \end{cases} \quad , \quad p_{z|x}(c|a) = \begin{cases} 1/N_{z|x} & , c \in T_{z|x}(a) \\ 0 & , c \notin T_{z|x}(a) \end{cases}$$



# Random binning encoding

If we can find:

- ▶ a subset  $\mathcal{X}' \subset \mathcal{X}$  such that  $\forall a \neq a' \in \mathcal{X}', T_{y|x}(a) \cap T_{y|x}(a') = \emptyset$
- ▶ a message space  $\mathcal{M}$  and a partition of  $\mathcal{X}'$  into  $\{T_{x|u}(d)\}_{d \in \mathcal{M}}$  such that  $\cup_{a \in T_{x|u}(d)} T_{z|x}(a) = \mathcal{Z}$ ,  $\forall d \in \mathcal{M}$



# Perfect reliability

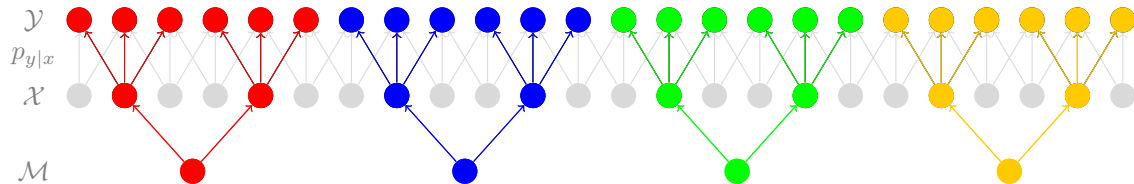
## Theorem

Let  $p_{yz|x}$  be a uniform wiretap channel and  $\mathcal{M}$  the message space for secret transmission over it.  
If:

- ▶  $\exists \mathcal{X}' \subset \mathcal{X}$  such that  $\forall a \neq a' \in \mathcal{X}', T_{y|x}(a) \cap T_{y|x}(a') = \emptyset$
- ▶  $\exists$  a collection  $\{T_{x|u}(d)\}_{d \in \mathcal{M}}$  of subsets of  $\mathcal{X}'$  such that  $\forall d \neq d' \in \mathcal{M}, T_{x|u}(d) \cap T_{x|u}(d') = \emptyset$
- ▶ the random encoder satisfies  $p_{x|u}(a|d) = 0, \forall a \notin T_{x|u}(d)$

then there exist a decoding rule that achieves perfect reliability

## Illustration



# Perfect reliability

## Proof.

Let  $T_{y|u}(d) = \cup_{a \in T_{x|u}(d)} T_{y|x}(a)$  be the subset of  $\mathcal{Y}$  reachable from each  $d \in \mathcal{M}$ . Since the  $T_{y|u}(d)$  are all disjoint, we can define the decoder

$$\hat{u} = d \quad , \quad \text{if } y \in T_{y|u}(d)$$

and we can compute the probability of correct detection as

$$\begin{aligned} \mathbb{P}[\hat{u} = u] &= \sum_{d \in \mathcal{M}} \mathbb{P}[\hat{u} = d | u = d] p_u(d) \\ &= \sum_{d \in \mathcal{M}} \mathbb{P}[y \in T_{y|u}(d) | u = d] p_u(d) = \sum_{d \in \mathcal{M}} p_u(d) = 1 \end{aligned}$$



# Perfect secrecy

## Theorem

Let  $p_{yz|x}$  be a uniform wiretap channel and  $\mathcal{M}$  the message space for secret transmission over it.  
If:

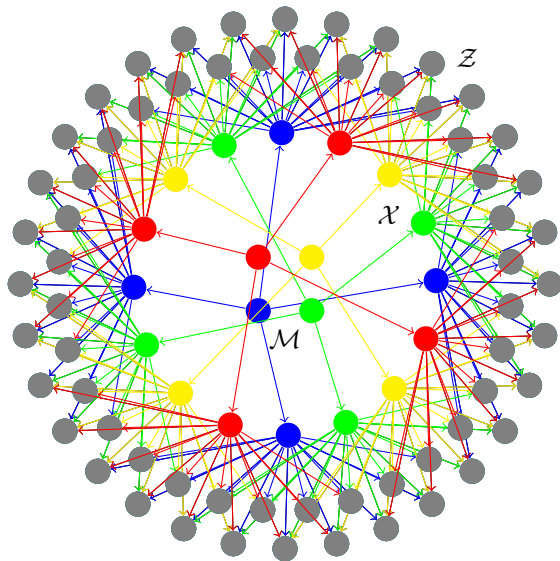
- ▶  $\exists$  a collection  $\{T_{x|u}(d)\}_{d \in \mathcal{M}}$  of subsets of  $\mathcal{X}$  such that by letting  $\mathcal{X}_{d \rightarrow c} = \{a \in \mathcal{X} : a \in T_{x|u}(d), c \in T_{z|x}(a)\}$  it holds  $|\mathcal{X}_{d \rightarrow c}| = N$ ,  $\forall c \in \mathcal{Z}, d \in \mathcal{M}$  for some  $N$  same for all  $c$  and  $d$
- ▶ the random encoder satisfies

$$p_{x|u}(a|d) = \begin{cases} 1/N_{x|u} & , a \in T_{x|u}(d) \\ 0 & , a \notin T_{x|u}(d) \end{cases}$$

then we have perfect secrecy of  $u$  wrt  $z$

# Perfect secrecy

Illustration for  $N = 1$





# Perfect secrecy

## Proof.

We show that  $u$  and  $z$  are independent. In fact:

$$\begin{aligned} p_{z|u}(c|d) &= \sum_{a \in \mathcal{X}} p_{z|x|u}(c|a, d) p_{x|u}(a|d) \\ &= \sum_{a \in \mathcal{X}_{d \rightarrow c}} p_{z|x}(c|a) p_{x|u}(a|d) \\ &= N \frac{1}{N_{z|x}} \frac{1}{N_{x|u}} \end{aligned}$$

which is independent of the particular value  $d$  of  $u$  (and also uniform wrt  $c \in \mathbb{Z}$ ) □

# How many secret bits can be sent?

For perfect **reliability** to B:

$$|\mathcal{X}'| \leq \frac{|\mathcal{Y}|}{N_{y|x}}$$

For perfect **secrecy** with respect to E:

$$N_{x|u} \geq \frac{|\mathcal{Z}|}{N_{z|x}}$$

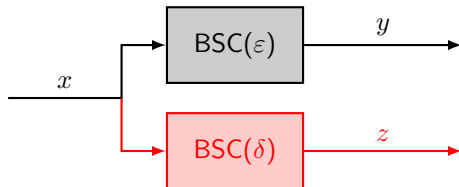
For both **reliability** and **secrecy**:

$$M = |\mathcal{M}| \leq \frac{|\mathcal{X}'|}{N_{x|u}} \leq \frac{|\mathcal{Y}|}{N_{y|x}} \frac{N_{z|x}}{|\mathcal{Z}|}$$

Secret bits in one channel use:  $\log_2 M$

# Secrecy capacity for the wiretap BSC

Let the channels from A to B and from A to E be memoryless binary symmetric with error rates  $\varepsilon$  and  $\delta$ , respectively



If  $|\varepsilon - \frac{1}{2}| < |\delta - \frac{1}{2}|$   
legitimate channel is degraded  
(e.g.,  $0 < \delta < \varepsilon < \frac{1}{2}$ )

$$C_s = 0$$

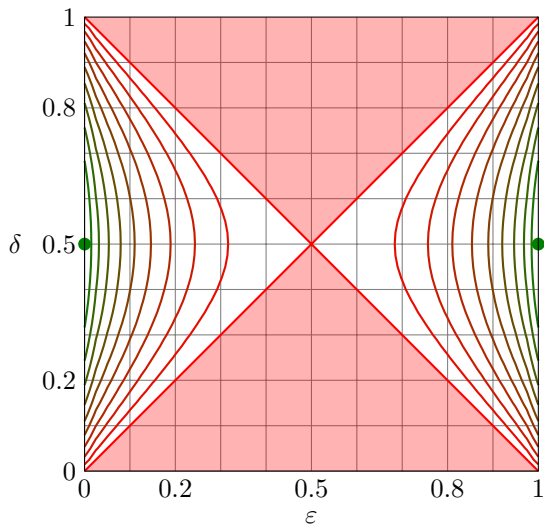
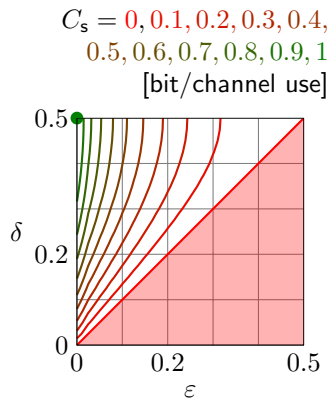
no secrecy is possible

If  $|\varepsilon - \frac{1}{2}| > |\delta - \frac{1}{2}|$   
eavesdropper channel is degraded  
(e.g.,  $0 < \varepsilon < \delta < \frac{1}{2}$ )

$$C_s = C_{AB} - C_{AE} = h_2(\delta) - h_2(\varepsilon)$$

where  $h_2(\varepsilon) = \varepsilon \log_{1/2} \varepsilon + (1 - \varepsilon) \log_{1/2} (1 - \varepsilon)$

# Secrecy capacity for the wiretap BSC



# Implement the uniform error channel

Consider a uniform error wiretap channel, where

- ▶ the input and output alphabets are the set of 7-bit words,  $\mathcal{X} = \mathcal{Y} = \mathcal{Z} = \{0, 1\}^7$
- ▶ the legitimate channel introduces at most 1 binary error per word,  

$$T_{y|x}(a) = \{b \in \mathcal{Y} : d_H(a, b) \leq 1\}$$

$$= a \oplus \{0000000, 0000001, 0000010, 0000100, 0001000, 0010000, 0100000, 1000000\}$$
- ▶ the eavesdropper channel introduces at most 3 binary error per word,  

$$T_{z|x}(a) = \{c \in \mathcal{Z} : d_H(a, c) \leq 3\}$$

$$= a \oplus \{0000000, 0000001, \dots, 1101000, 1110000\}$$
- ▶  $y$  and  $z$  are conditionally uniform and independent of each other given  $x$

## Task 1

Using a programming language of your choice, implement the wiretap channel, so that it accepts an input  $x \in \mathcal{X}$  and produces the corresponding pair of outputs  $(y, z)$

Verify the conditional independence and uniformity of your outputs by running a sufficiently large number (at least  $10^4$ ) of channel realizations, with the same input  $x$  (e.g.,  $x = 1001000$ ) and gathering statistics

# Implement the random binning encoder

Consider a uniform binning encoder, where

- ▶ the code is the  $(7, 4)$  Hamming code,  
 $\mathcal{X}' = \{0000000, 1000110, 0100101, 0010011, 0001111, 1100011, 1010101, 1001001, 0110110, 0101010, 0011100, 1110000, 1101100, 1011010, 0111001, 1111111\}$  (observe that the minimum Hamming distance in the code is 3)
- ▶ the message space is the set of 3-bit words,  $\mathcal{M} = \{0, 1\}^3$
- ▶ the bin  $T_{x|u}(d)$  associated to each input  $d \in \mathcal{M}$  is made of 2 codewords: the one having  $[0, d]$  as its 4-bit prefix, and the binary complement of the that codeword (e.g.,  $T_{x|u}(100) = \{0100101, 1011010\}$ )
- ▶ the codeword  $x$  is chosen randomly and uniformly within the bin associated to the message  $u$

## Task 2

Using a programming language of your choice, implement the random binning encoder, so that it accepts an input  $u \in \mathcal{M}$  and produces the corresponding output  $x \in \mathcal{X}'$

Verify the correctness of your implementation observing the codewords associated to different messages

# Implement the random binning decoder

Consider a deterministic legitimate decoder  $D : \mathcal{Y} \mapsto \mathcal{M}$ , which

1. identifies the transmitted codeword by the minimum Hamming distance criterion  

$$\hat{x} = \arg \min_{a \in \mathcal{X}'} d_H(y, a)$$
2. looks at the first bit of  $\hat{x}$  and identifies the transmitted message  $\hat{u}$  as either the bits 2-4 in  $\hat{x}$ , or their complement

## Task 3

Using a programming language of your choice, implement the legitimate decoder, so that it accepts an input  $y \in \mathcal{Y}$  and produces the corresponding output  $\hat{u} \in \mathcal{M}$

- ▶ Verify, by cascading encoder + decoder, that your decoder makes no errors. This is due to the property of the Hamming code being systematic.
- ▶ Verify, by cascading encoder + legitimate channel + decoder, that your decoder makes no errors. This is due to the property of the Hamming code being able to correct 1 error per word, however placed.

## Verify perfect secrecy

In order to prove that our encoder achieves perfect secrecy we must show that the eavesdropper channel output  $z$  is independent of the secret message  $u$

### Task 4

- ▶ Using a programming language of your choice, implement the encoder + eavesdropper channel chain, so that it accepts an input  $u \in \mathcal{M}$  and produces the corresponding output  $z \in \mathcal{Z}$ .
- ▶ For each possible value of  $u$ , simulate at least  $100 \cdot |\mathcal{Z}| \simeq 10^4$  realizations of the chain and gather the empirical distribution of  $z$
- ▶ plot or tabulate the empirical conditional pmf of  $z$  given  $u$ ,  $\hat{p}_{z|u}(c|d)$  for all values of  $d$
- ▶ compute the empirical joint  $\hat{p}_{u,z}(d, c)$  and marginal distributions  $\hat{p}_u(d)$ ,  $\hat{p}_z(c)$  and the mutual information

$$\hat{I}(u, z) = \sum_{d \in \mathcal{M}, c \in \mathcal{Z}} \hat{p}_{u,z}(d, c) \log_2 \frac{\hat{p}_{u,z}(d, c)}{\hat{p}_u(d) \hat{p}_z(c)}$$

Can you say that  $u$  and  $z$  are empirically independent within the statistical reliability of your simulations?



## Considerations and remarks

Ponder your work and answer the following questions:

1. How many secret message bits per channel use ( “transmitted word” ) have you obtained with your scheme?  
How many secret bits per binary digit ( “transmitted bit” )?
2. Is it possible to obtain 4 secret bits per channel use?  
If so, how should you change your encoder/decoder? If not, why?
3. Is it possible to obtain 2 secret bits per channel use?  
If so, how should you change your encoder/decoder? If not, why?
4. One could consider evaluating the secrecy of this mechanism by cascading the eavesdropper channel with a decoder and measuring the resulting error rates. What do you expect Eve’s error would be?  
Why resort to (more complicated) evaluating the mutual information?

## Simulate transmission over a binary symmetric channel

Consider a wiretap binary symmetric channel, with independent errors on the two branches, and error rates  $\varepsilon$  and  $\delta$  for the legitimate and eavesdropper channel, respectively.

### Task 5

Using a programming language of your choice, implement the wiretap BSC, so that

- ▶ it can be simulated with arbitrary values of  $\varepsilon$ ,  $\delta$ ,
- ▶ it can be connected in between the random binning encoder developed in Task 2, and the decoder developed in Task 3

Verify the correctness of your implementation by transmitting a long binary sequence and checking the number of bit errors in each output.

Connect the wiretap channel to the random binning encoder and the legitimate decoder, and simulate several transmissions.

Observe that perfect reliability or secrecy are no longer provided, as there may be more than one error in a single codeword over the legitimate channel, and the error pattern distribution in the eavesdropper channel is no longer uniform.

# Evaluate the system security over the wiretap BSC

## Task 6

Choose several values of  $\varepsilon$  and  $\delta$ , and for each  $(\varepsilon, \delta)$  pair

- ▶ repeat the simulations in Tasks 3-4 with the wiretap BSC
- ▶ evaluate the resulting reliability in terms of Bob's error rate on the secret message  $P[u \neq \hat{u}]$
- ▶ evaluate the resulting the secrecy in terms of leaked information to Eve on the secret message  $I(u; z)$
- ▶ compute an upper bound to the mechanism security in terms of distinguishability from the ideal counterpart

# What you need to turn in

Each team must turn in, through the Moodle assignment submission procedure:

1. the source code for your implementation (either as a single file, many separate files, or a compressed folder)
2. a short report (to be submitted as a separate file from the source code file / compressed folder) in a graphics format (PDF, DJVU or PostScript are ok; Word, T<sub>E</sub>X or L<sup>A</sup>T<sub>E</sub>X source are not), including:
  - 2.1 a brief description of your implementations for Tasks 1-6, explaining your choices;
  - 2.2 your answers to the questions in **Considerations and remarks**
  - 2.3 the evaluated security metrics for your system:
    - 2.3.1 a plot of the conditional pmd  $p_{z|x}(\cdot|1001000)$  from Task 1;
    - 2.3.2 the plots of the conditional pmfs  $p_{z|u}(\cdot|d)$  for all values of  $d$  from Task 4;
    - 2.3.3 the estimates of  $H(u)$ , and  $I(u; z)$  from Task 4
    - 2.3.4 a plot of the error decoding probability  $P[\hat{u} \neq u]$  as a function of  $\varepsilon$  for the BSC from Task 6
    - 2.3.5 a plot of the mutual information  $I(u; z)$  as a function of  $\delta$  for the BSC from Task 6
    - 2.3.6 a contour plot of the security of this mechanism as a function of  $\varepsilon$  and  $\delta$  for the BSC from Task 6

## Appendix: Hamming codes

Hamming codes are **linear binary codes** with the following properties:

1. There **exist** Hamming codes with dimension  $k$  and length  $n$  (i.e., with  $2^k$  codewords of  $n$  bits) **if and only if**  $n + 1 = 2^{n-k}$
2. A **generating matrix** for a Hamming code of length  $n$  is given by  $G = \left[ \frac{I_k}{A} \right]$  where  $I_k$  is the  $k \times k$  identity matrix and the  $k$  columns of  $A$  are all the words of  $n - k$  bits with Hamming weight  $\geq 2$
3. The **minimum Hamming distance** between any two codewords is  $d_{\min} = 3$
4. Every binary word in  $\{0, 1\}^n$  is either **a codeword or at distance 1** from a codeword

As a consequence of the above properties 3-4, when used in forward error correction over a binary channel:

- ▶ a Hamming code can **exactly correct 1 error** in each codeword, however placed;
- ▶ a Hamming code has the **highest rate** (i.e., maximum number of codewords, maximum information carried) among all the single-error correcting codes of the same length  $n$ .