Faculty of Information Engineering & Technology
The Communications Department
Course: Channel Coding [COMM 604]
Spring 2021

Practical Assignment

Consider a channel encoder/decoder system. Assume the information bits (i.e., bits input to the channel coding process) are extracted from a video stream. The channel coded bits are transmitted over a communication channel with probability of error p. You are asked to write a MATLAB code to simulate the transmission of the encoded bits over the communication channel using an incremental redundancy system as follow:

Code Rate	Puncturing Rule	Upgrading Code
8/9	X:1111 0111 Y:1000 1000	X:1111 0111 Y:1000 1000
4/5	X:1111 **1**111 Y:1000 1000	X:0000 **1**000 Y:0000 0000
2/3	X:1111 1111 Y:10**1**0 10**1**0	X:0000 0000 Y:00**1**0 00**1**0
4/7	X:1111 1111 Y:1**1**10 1**1**10	X:0000 0000 Y:0**10**0 0**10**0
1/2	X:1111 1111 Y:111**1** 111**1**	X:0000 0000 Y:000**1** 000**1**

1. The Video stream is represented as a binary sequence.

2. The binary sequence representation of the video stream is divided into messages of size 1024 each.

3. Each message is encoded with a rate 1/2 mother convolutional code with the generators 133 and 171 in octal form. (for a rate- 1/2 packet size of 2048 bits).

4. The 2048 bits (rate-1/2 packet) is punctured to become a rate- 8/9 packet (i.e., not transmitting 7 bits from every 16 bits generated by the rate 1/2 code) using the puncturing pattern in Table A. The rate-8/9 packet size is 1152 bits.

5. The rate-8/9 packet is then transmitted over a BSC channel with error probability p.

6. The received packet is corrected by a Viterbi decoder in accordance to the 8/9 code rate.

7. The corrected message (1024 bits) is compared with the original transmitted message (1024 bits).

a. If they are the same then the message is assumed to be correct and the next 1024 bits message from the video stream is dealt with.

b. If they are not the same then an error is assumed and the transmitter must upgrade to the next rate which is 4/5.

8. The upgrade to rate-4/5 packets (rate 8/10) necessitates a packet size of 1280 bits of which 1152 bits have been already transmitted in the rate-8/9 packet. Accordingly the upgrade in incremental redundancy means that only the additional bits (1280-1152=128 bits) are applied to the BSC.

9. The combined rate-4/5 packet (1280 bits) is corrected by a Viterbi decoder.

10. The corrected message (1024 bits) is compared with the original transmitted message (1024 bits).
a. If they are the same then the message is assumed to be correct and then next 1024 bits message from the video stream is dealt with.
b. If they are not the same then an error is assumed and the transmitter must upgrade to the next rate which is 2/3.

11. The process is repeated from rate 2/3 to rate 4/7 to finally rate 1/2.

12. If at the rate 1/2 message is still in error after all possible code upgrades are completed then the message is accepted as it is and the next 1024 bits message is dealt with.

NOTES:
1. You are allowed to use MATLAB built in functions for the encoder and decoder.
2. You are encouraged to work in teams that SHOULD NOT exceed 5 students (group members do not have to be in the same tutorial).

Project summary
Each group should submit a MATLAB code that:

- reads an .avi file
- converts the file to bits
- subdivides the video stream to packets of length 1024
- encodes packets using the convolutional code is step 3
- decodes using the same sequence using Viterbi decoder
- reconstructs the video stream
- saves the corresponding video file

<u>Each group should submit the following in a compressed folder:</u>

- A SINGLE document with the following content:
  - Curves that reflect the following:
    - Plot of the coded bit error probability vs. different values of p from (0.0001 to 0.2) assuming code rate =1/2.
    - Plot of the coded bit error probability vs. different values of p from (0.0001 to 0.2) using incremental redundancy (increasing code rate).
    - Plot of the throughput (data rate) vs. different values of p from (0.0001 to 0.2) using incremental redundancy.
- Commented Matlab code (You must explain what you are doing).
- Six video files for the decoded video:
  1. P=0.001 using no channel coding
  2. P=0.001 using rate 1/2 convolutional code
  3. P=0.001 using incremental redundancy
  4. P=0.1 using no channel coding
  5. P=0.1 using rate 1/2 convolutional code
  6. P=0.1 using incremental redundancy

<u>You will submit your project by sending the compressed folder to the following e-mail address:</u>

Spring.guc.2021@gmail.com

Project submission deadline is on 15/6/2021

<u>Project Evaluations</u>

Individual project Evaluation tasks will take place in the week after submissions. You will be notified with the exact locations and timings.

Any similar projects will be assigned zeros.