



LECTURE CONTENT: DIGITAL VIDEO BROADCASTING – TERRESTRIAL (DVB-T)

- Video communication systems
 - Channel Encoding
 - DVB-T
 - Video error concealment
 - Matlab-Simulink model
- Using the graphical user interface
 - DVB-T GUI
- Video Error Concealment GUI

Lecture 6





DIGITAL VIDEO BROADCASTING - TERRESTRIAL (DVB-T)

1. BRIEF THEORY

1.1. INTRODUCTION - VIDEO TRANSMISSION OVER NOISY CHANNELS

Generally, digital compressed video is considerably vulnerable to transmission errors, due to the use of predictive and variable-length (entropy) coding. Even a single bit error can lead to damage of a whole block or set of video blocks. In contrast with simple data communications, which are not subject to strict delay conditions, real-time video transmission is delay sensitive and cannot make use of retransmission in order to ensure error-free delivery of the video data. Therefore, successful video communication in the presence of errors requires careful design of the systems.

1.1.1. Video communication systems

A general video communication system can be described by the diagram of Fig. 1.

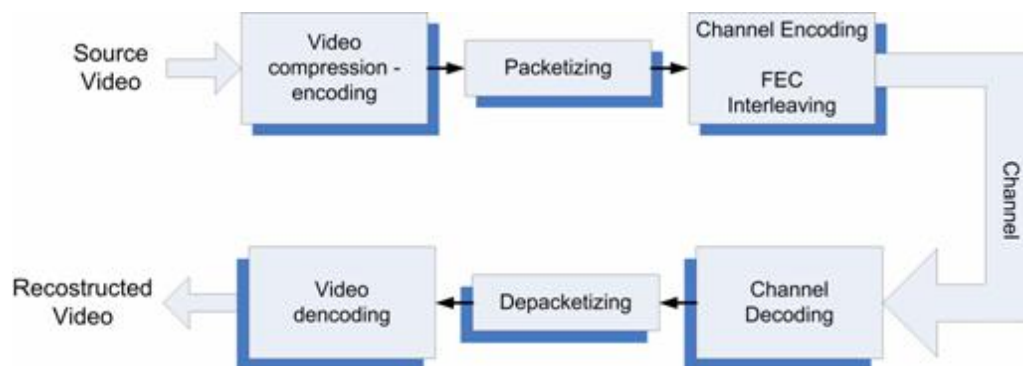


Figure 1. The general structure of a typical video communication system

For effective video communications, reduction of raw video rates, i. e. video coding, is only the first of the necessary steps. The compressed bit stream is segmented into fixed or variable length packets. The packets could be sent directly over the channel, if the channel guaranteed error-free transmission. However, this is not the case. Therefore, the packets usually undergo a Channel Coding stage, using Forward Error Correction (FEC) techniques and Interleaving, in order to become more robust against transmission errors. At the receiver, the received packets are Channel Decoded and unpacked, before the resulting bit stream is decompressed.

Transmission errors can be roughly categorized into a) random bit errors and b) erasure errors. Erasure errors are mainly caused in packet networks, such as the Internet. In video communication systems, such as the Digital Video Broadcasting - Terrestrial (DVB-T) system, we will concentrate mainly in random bit errors. The use of Variable Length Coding (VLC) (e.g. Huffman) makes compressed video vulnerable to random bit errors, because a bit error can desynchronize the coded information and many following bits are wrongly decoded or become undecodable.

Due to the above reason and the fact that video compression techniques use predictive coding (e.g. a frame is predicted from a previous one in MPEG), error control in video communications is a very challenging task. Because of VLC, a single error can cause damage over a large portion of a video frame. Furthermore, due to the use of temporal prediction, a single erroneously recovered sample can lead to errors in future frames, i.e. the error propagates temporally.

1.1.2. Channel Encoding

Forward Error Correction (FEC)

In order to render the bit stream more resilient to bit errors, the transmitter must add redundancy (i.e. some additional bits) to the stream, so that error detection and correction to become possible. This is done using Forward Error Correction (FEC) techniques. However, notice that the addition of bit redundancy is opposite to the idea of video compression/coding which aims to the reduction of the bit rate. Therefore, generally, FEC must be carefully designed. Due to the same reason, the problem of joint video and channel coding is studied by the scientific community in the last years.

The mathematician R. Hamming pioneered the idea of FEC in the 1940's and invented the first FEC code, the Hamming (7,4) code in 1950. FEC is accomplished using a predetermined algorithm to add redundancy into the transmitted information. A redundant bit may be a complex function of many original information bits. The original information may or may not appear literally in the encoded output. FEC codes that include the original bits input in the output are systematic, while those that do not are non-systematic.

In order to understand how it works, consider that a very simple example of a FEC code could be to transmit each data bit 3 (e.g.) times. This is a simple (3,1) repetition code. An error in any one of the three samples could be corrected by a "majority vote" or "democratic voting". The correcting ability of such a FEC code is up to 1 error bit in the triplet.

There are two main categories of FEC codes:

- a. **Block codes:** This type of FEC codes works on fixed-length streams of bits. Practical block codes can generally be decoded in polynomial time to their stream length.
- b. **Convolutional codes:** Convolutional codes work on bit streams of arbitrary length. They are most often decoded with the Viterbi algorithm. Viterbi decoding allows asymptotically optimal decoding efficiency with increasing constraint length of the convolutional code, but at the expense of exponentially increasing complexity.

Classical block codes and convolutional codes can be combined in concatenated coding schemes. In this case, a short constraint-length Viterbi-decoded convolutional code does most of the work. The block code (e.g. a Reed-Solomon code), with a larger block length, detects and corrects any errors made by the convolutional decoder.

Interleaving

Interleaving is a way to rearrange digital data in a non-contiguous way and it is frequently used in digital communication systems in order to improve the performance of Forward Error Correcting codes.

In many cases of communication channels, errors typically occur in bursts rather than independently. Therefore, if the number of errors within a code word exceeds the error-correcting code's capability, the latter fails to recover the original code word. Interleaving helps us to override this problem by shuffling the source bits across several code words, thereby creating a more uniform distribution of errors [3].

1.2.DVB-T

Digital Video Broadcasting - Terrestrial, abbreviated as "DVB-T" is a European-based consortium standard for the broadcast transmission of digital terrestrial television. DVB-T transmits compressed digital audio and video in an MPEG

transport stream, using coded Orthogonal Frequency-Division Multiplexing (OFDM) modulation.

With respect to Fig. 2, a coarse and short description of the DVB-T system follows. Notice that the diagram of Fig. 2 is in agreement with general video transmission diagram of Fig. 1.

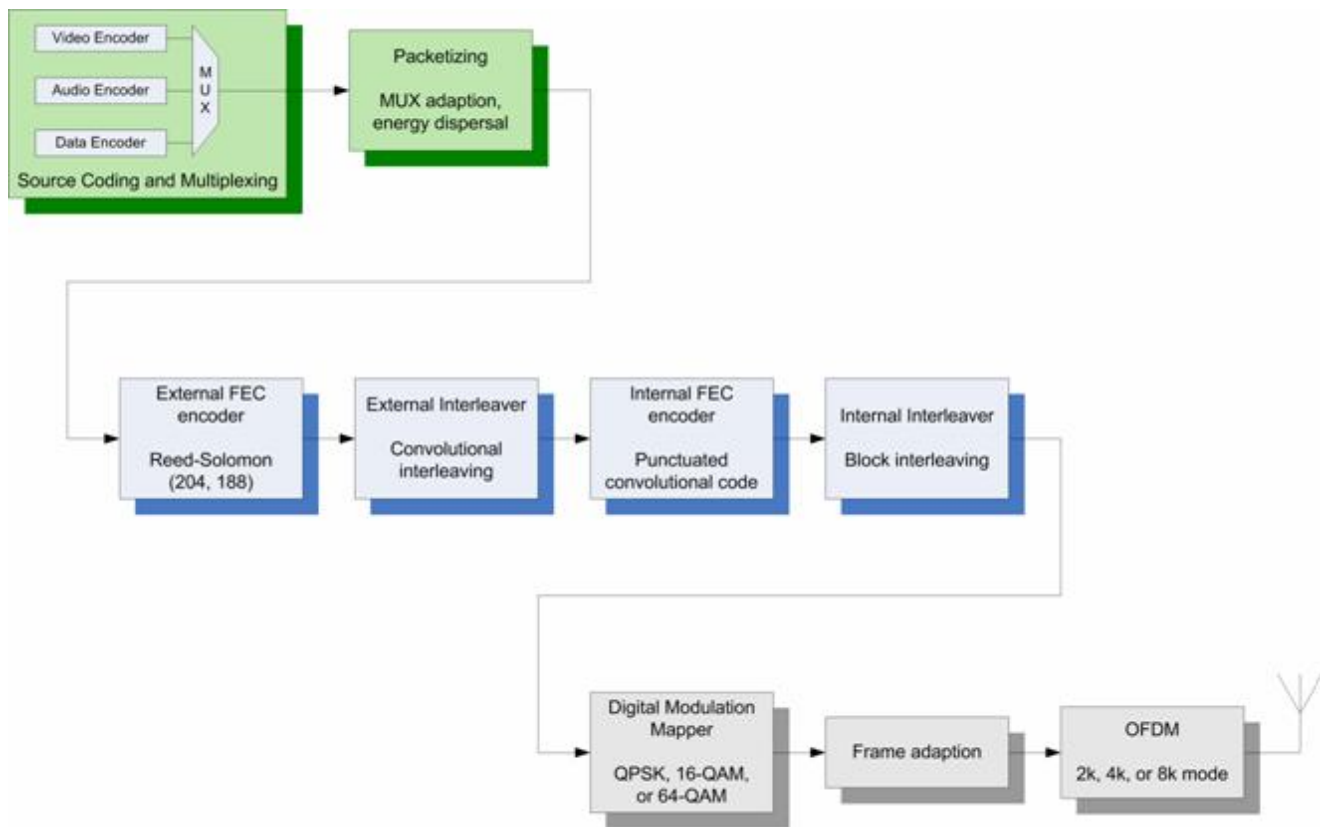


Figure 2. A coarse description diagram of a DVB-T transmission system

- **Source coding and multiplexing:** Source video and audio are compressed. Compressed video, audio, and data streams are multiplexed into MPEG program streams. One or more program streams are joined together into an MPEG transport stream, which is the basic digital stream transmitted and received by our TV sets.
- **Packetizing:** The MPEG transport stream is split into a sequence of 188-bytes-long packets. The byte sequence is decorrelated using an “energy dispersal” technique.
- **External FEC encoder:** A first level of FEC encoding is applied to the transmitted data. A non-binary block code, a Reed Solomon RS (204, 188) code is used. This code allows the correction of up to a maximum of 8 wrong bytes for each 188-byte packet.
- **Internal interleaver:** An internal interleaver is used to rearrange the data sequence. A convolutional interleaver is used. The rearrangement is such that the transmitted data becomes more resilient to long sequences of errors.
- **Internal FEC encoder:** A second level of FEC error correction is applied. A punctuated convolutional code is used. There are five valid coding rates for DVB-T: 1/2, 2/3, 3/4, 5/6, and 7/8.
- **Internal interleaver:** An internal block interleaving technique is adopted to rearrange the data sequence again. This inner interleaver aims to reduce the influence of burst errors.
- **Digital modulation mapper:** QPSK, 16-QAM, or 64-QAM is used to map the bit sequence into a (base-band) sequence of complex symbols.

- **Frame adaption:** The complex symbols are grouped in blocks of constant length. A sequence of 1512, 3024, or 6048 symbols constitutes a block.
- **OFDM modulation:** The sequence of blocks is modulated using the OFDM technique, DVB-T uses 2048, 4096, or 8192 OFDM subcarriers (2k, 4k, 8k mode, respectively).

1.3.VIDEO ERROR CONCEALMENT

Despite the use of channel coding techniques, uncorrected transmission errors often occur, which may lead to objectionable visual distortions in the reconstructed video at the decoder.

When a video block is damaged or missing due to transmission errors, the decoder can estimate it based on surrounding received and correct blocks. This is a reasonable idea, based on the inherent spatial and temporal correlation among the video samples. Such techniques are known as “error concealment” techniques. Error concealment, in contrast to FEC coding, has the advantage of not employing any additional bits. However, error concealment increases the complexity of the decoder.

There are two main types of information that may need to be estimated in a damaged MPEG MacroBlock (MB): a) The texture information, i.e. the pixel values (or the DCT coefficient values) for either an original block (I-frame mode) or a prediction error block (P- or B -frame mode), b) The motion information, consisting of Motion Vectors for a P- or B-mode MacroBlock. The recovery techniques for these information types are somewhat different.

It is well known that for natural (moving) images, the energy is concentrated in low frequency components, i.e. the intensity and color values of spatially (and temporally) adjacent pixels vary smoothly, except in regions with edges. Therefore, the techniques that are used for texture information recovery make use of this smoothness property. They all perform some kind of spatial and/or temporal interpolation.

Spatial interpolation

The simplest approach to recover a damaged video block is to interpolate the damaged pixels from pixels in adjacent correctly received blocks of the same frame. However, the pixels in the center of the block are far away from the pixels of the correct surrounding blocks. Therefore, spatial interpolation gives satisfactory results only for smooth regions of the image.

Motion compensated temporal prediction

Another simple but slightly more efficient approach to recover a damaged video block is to copy the corresponding block in a previously decoded frame. This approach is implemented in the IHU Virtual Lab Simulink model for video error concealment. However, it can produce satisfactory results only for static or slowly moving regions of the image. For regions of large motions a more effective solution is to use the block in the previous frame pointed to by the MV for the damaged block.

The reader can find more details about error concealments techniques in [1, pp. 500].

1.4.REFERENCES

- [1] Y. Wang, J. Ostermann, Y. Zhang, *Video Processing and Communications*, New Jersey: Prentice Hall, 2002.
- [2] Wilson, Stephen G. *Digital Modulation and Coding*, Englewood Cliffs, New Jersey: Prentice-Hall, 1996
- [3] B. Vucetic, J. Yuan. *Turbo codes: principles and applications*, Springer Verlag, 2000.
- [4] ETSI Standard EN 300 744: *Digital Video Broadcasting (DVB); Framing structure, channel coding and modulation for digital terrestrial television*, European Telecommunications Standards Institute, Valbonne, France, 1997.





DIGITAL VIDEO BROADCAST - TERRESTRIAL (DVB-T)

2. “DVB-T” USER GUIDE

The Graphical User Interface (GUI), which is illustrated in Fig. 1, is used for the simulation of MPEG video compression and transmission over noisy channels. Specifically, the simulation is performed with the use of a Matlab-Simulink model, which compresses the input video according to the MPEG standard and models a part of the ETSI (European Telecommunications Standards Institute) EN 300 744 standard for terrestrial transmission of digital television signals (Digital Video Broadcast Terrestrial, DVB-T). The main part of the model and the DVB-T channel encoding-transmission-decoding subsystem are given in Figs. 2-3. The GUI helps the user to change the model's parameters, during the simulation.

With the use of this demo the user can study:

- The effects of transmission errors to the quality of the decoded video.
- The idea of Forward Error Correction (FEC) / Channel Coding techniques.

The user can select any input video. Using the selected video, the model simulates:

- The MPEG compression, with all of its components:
 - Block DCT transform,
 - Block Matching Motion Estimation,
 - Quantization and entropy coding.
- The Channel coding operations, applied by a DVB-T transmitter:
 - Reed Solomon coding,
 - Convolutional encoding,
 - Interleaving, etc.
- Modulation according to DVB-T: 64-QAM and 2-K OFDM.
- Transmission over Additive White Gaussian Noise (AWGN) channel.
- Demodulation.
- Channel Decoding.
- MPEG decoding.

The user can view the input video, the compressed one and the video at the receiver. Furthermore, the user can study the (baseband) OFDM spectrum of the received signal.

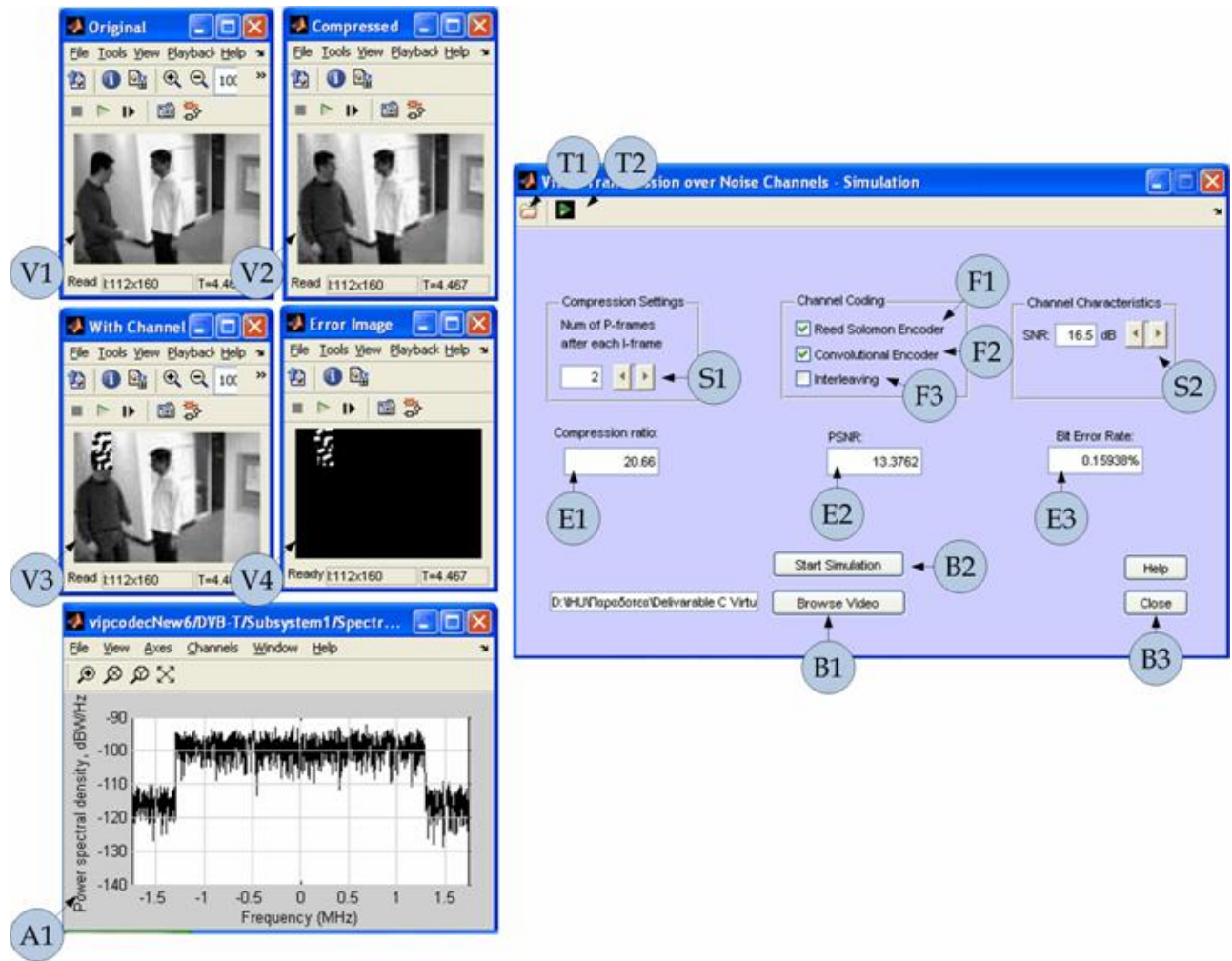


Figure 1. A snapshot of the “DVB-T” GUI. The used notation is explained in the text.

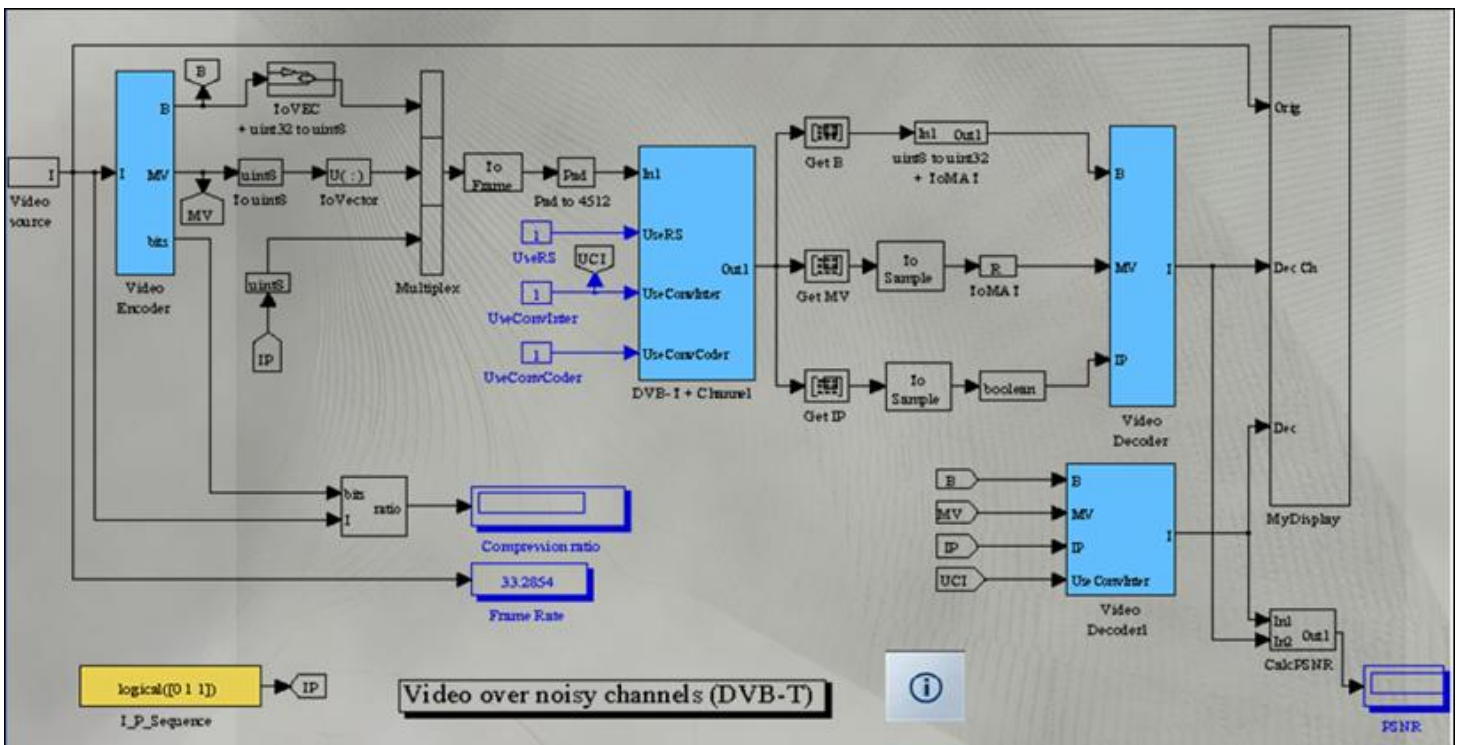


Figure 2. The main part of the DVB-T model.

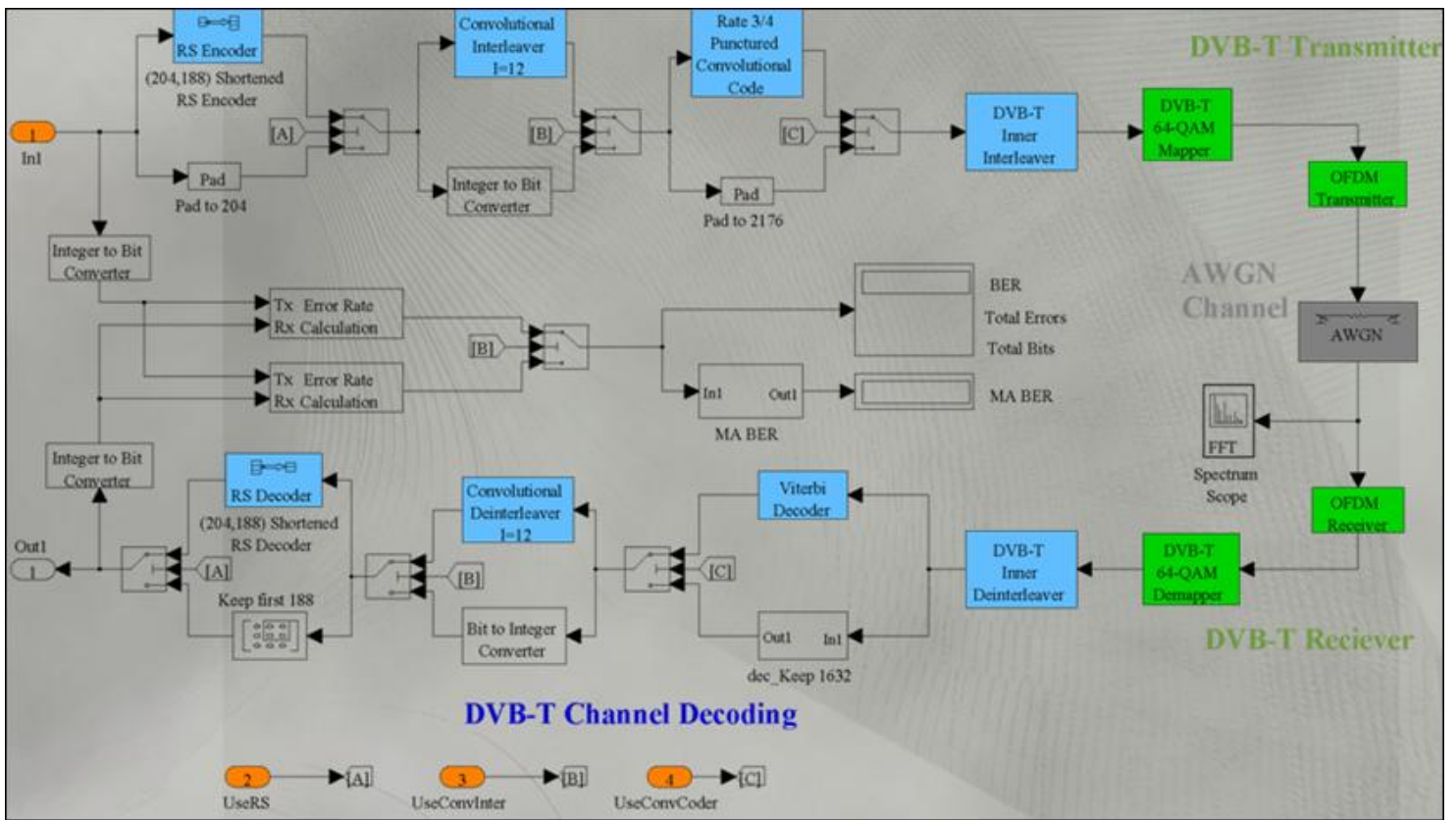


Figure 3. The DVB-T channel encoding-transmission-decoding subsystem.

2.1. NOTATION

The notation that is used in Fig. 1 is as follows:

- T_i : Toolbar button
- E: Edit Box
- S: Slider
- F: Flag (CheckBox)
- B: Button
- V: Video panel
- A: Axes

2.2. USING THE GRAPHICAL USER INTERFACE

The user selects an input video using the toolbar button T_1 or button B_1 .

The simulation starts with the use of button B_2 or toolbar button T_2 . During simulation, the input video is displayed in the Video panel V_1 . The compressed one is given in the Video panel V_2 , while the received and decoded video is shown in V_3 . The difference of the transmitted video and the received one is given in V_4 . Finally, the (baseband) OFDM spectrum of the received signal is shown in the Axes panel A_1 .

Some important measures are presented in the Edit Boxes E_1 , E_2 and E_3 . The compression ratio, achieved by the MPEG encoder, is given in E_1 . The bit error rate due to the digital signal's transmission is displayed in E_3 . Finally, the PSNR (Peak Signal to Noise Ratio) that corresponds to the difference of the received and the transmitted video, is given in E_2 .

The user can change the parameters of the model, during the simulation. Specifically, using the Slider S_1 the user can specify the number of P-frames after each I-frame (see the MPEG section), considered in the MPEG compression stage. Using the Slider S_2 , the user can change the AWGN level of the transmission channel and specifically the corresponding SNR (Signal to Noise Ratio). Finally, using the CheckBoxes F_1 , F_2 and F_3 , one can specify whether the transmitter applies Reed-Solomon encoding, Convolutional encoding and interleaving, respectively.

The user can close the GUI and the Simulink model using the Button B_3 .

Two examples of this GUI's usage are given Figs. 4 and 5, together with conclusions that can be extracted.

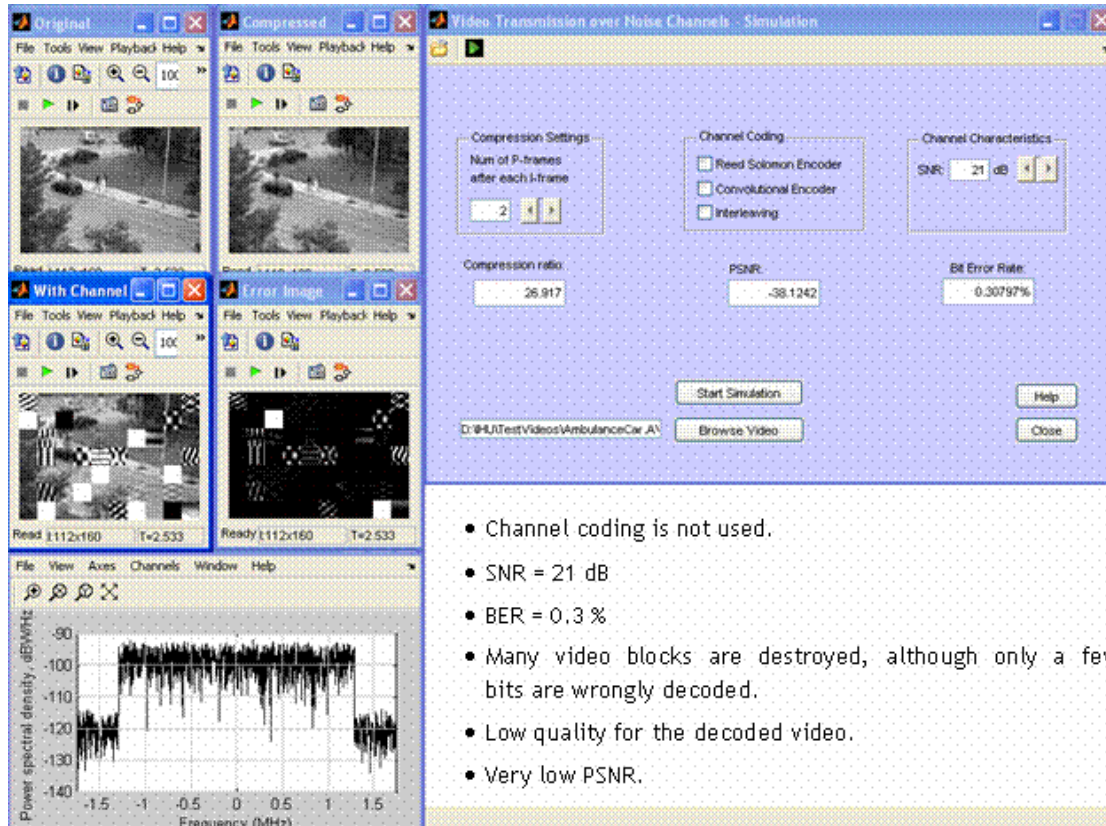
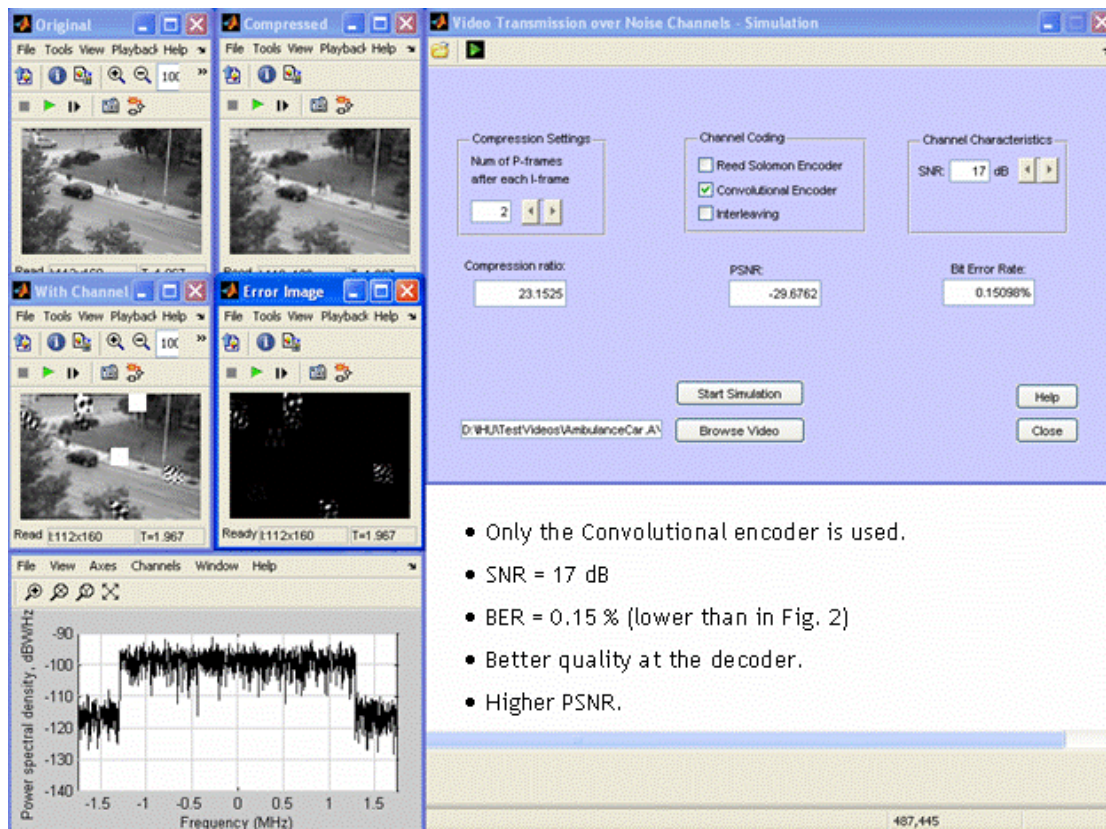


Figure 4. An example of the GUI's usage and some conclusions.



- Only the Convolutional encoder is used.
- SNR = 17 dB
- BER = 0.15 % (lower than in Fig. 2)
- Better quality at the decoder.
- Higher PSNR.

Figure 5. A second example of the GUI's usage and some conclusions.





ERROR CONCEALMENT IN DIGITAL VIDEO TRANSMISSION

4. “VIDEO ERROR CONCEALMENT” USER GUIDE

The “Video Error Concealment” Graphical User Interface (GUI) is given in Fig. 1. This GUI is used for the simulation of a simple Video Error Concealment technique that could be used in a DVB-T receiver. The simulation is performed with the use of a Simulink model, which constitutes a simplification of the DVB-T model (see “DVB-T” User Guide section). An “error concealment” block was added to the MPEG decoder of the model. This block is illustrated in Fig. 2. This block takes input (see input “4: Err”) from the Reed-Solomon decoder block, which can detect the non-corrected bit errors. Based on this input, the destroyed video blocks are detected and replaced by the corresponding previous frame’s blocks.

With the use of this demo the user can:

- Study the idea of “Error concealment” at the decoder.
- Observe the improvement of the video quality, using error concealment techniques.

The notation that is used in Fig. 1 is as follows:

- **T₁**: Toolbar button
- **E**: Edit Box
- **S**: Slider
- **F**: Flag (CheckBox)
- **B**: Button
- **V**: Video panel
- **A**: Axes

4.1. USING THE GRAPHICAL USER INTERFACE

The user can select the input video using the toolbar button **T₁** or button **B₁**.

The model’s simulation begins with the use of buttons **B₂** or **T₂**. During simulation, the input video is displayed in the Video panel **V₁**. The compressed video is displayed in the Video panel **V₂**. The received video without error concealment is shown in **V₃**. The error concealment results are shown in the Video panel **V₄**. The difference of the corrected video (video after error concealment) from the transmitted one is displayed in **V₅**. Finally, the (baseband) OFDM spectrum of the received signal is shown in the Axes panel **A₁**.

Figure 2. The “Error Concealment” subsystem of the model.

The compression ratio, achieved by the MPEG encoder, is given in the Edit Box E_1 . The bit error rate is displayed in E_3 . Finally, the PSNR (Peak Signal to Noise Ratio) that corresponds to the difference of the corrected video from the transmitted one, is given in E_2 .

Using the Slider S_1 the user can specify the number of P-frames after each I-frame (see the MPEG section), considered in the MPEG compression stage. Using the Slider S_2 , one specifies the SNR (Signal to Noise Ratio) for the transmission channel. Finally, using the CheckBoxes F_1 , and F_2 , one can specify whether the transmitter applies Convolutional encoding and interleaving, respectively. The Reed-Solomon coder/decoder is required for the bit-errors detection. One can change the parameters of the model, during the simulation.

The user can close the GUI and the Simulink model using the Button B_3 .

Co-funded by the
Erasmus+ Programme
of the European Union

