Channel Encoding & Decoding

Viterbi Algorithm

Stefan Kroll
Christoph Kaiser
Roland Ortner

Martin Schober
Peter Mödlhammer
Hannes Graf
Channel Encoding & Decoding

1. Digital Communication System
2. Viterbi Algorithm
3. Advantages of the Viterbi Algorithm
4. Presentation of a Demonstration Software
1. Digital communication system

- Definition of Viterbi algorithm
- Historical background of viterbi-coding
- Parts of a digital communication system
- Use of the Viterbi algorithm
Digital communication system

• Problem of digital communication
  – Transmit much data via a noisy channel
  – Detect and correct errors

• Solution
  – Convolutional coding with
  – Forward error correction (FER)
Viterbi Algorithm

The Viterbi-Algorithm is a basic part of the coding and modulation method of the digital data transmission.

With the help of the Viterbi-algorithm it is possible to recognise data errors and correct them at the receiver. [9]
Viterbi Algorithm

"...efficient method of optimum sequence estimation of a finite-state process“. 

„finding the shortest path through a weighted graph“
Development

1. Digital communication system

< 1955

Channel Encoding was implemented by
  – Block codes
  – Linear block codes

1955

Elias P. introduced
  – Convolutional codes
  – Fixed decoding time
  – First real-time coding
Development

1967

Andrew J. Viterbi
– Founder of Qualcomm Corporation
– Developed the Viterbi Algorithm

Other researchers improved the channel coding methods
– Found new convolutional codes
– Improved the performance limits
– Designed implementations in hard- & software
Development

Turbo coding merged

- Parallel concatenated convolutional technique
- Improves performance by chaining up: Viterbi decoder and Reed-Solomon decoder
  (data recycle through the decoder several times)

Viterbi & Reed-Solomon decoding

- Used in space communication
- Geostationary satellite communication
Digital communication system

1. Information source
   - Speech of a person
   - Binary data of a machine e.g. computer
   => Analog wave / sequence of discrete symbols
Digital communication system

2. **Source encoder** (A/D converter)
   - Transformation of analogue signal into digital signal
   - Compression of signal
   => Information sequence
3. **Channel encoder**

- Transformation of information sequence into encoded sequence
- Adding of redundant information for error recognition and correction
  => Code word
Digital communication system

4. Modulator
• Modulation of a high frequency carrier by code word e.g. QPSK
5. Channel
Transmission via channel; Influence by noise
- Non-linearity of system
- Interferences in the atmosphere: rain, clouds, other radio waves
- Reflections
Digital communication system

6. Demodulator
- Detection of the phase
- Viterbi-decoding in real-time

=> Received sequence
Digital communication system

7. Channel decoder
   • Transformation of received sequence into a
     => Binary sequence
Digital communication system

8. **Source decoder (D/A converter)**
Transformation into a continuous waveform
=> Analogue signal (speech)
Channel coding: Code Types

Code Types:
- Block coding
- Convolutional coding
1. Block coding

- **Input:**
  - Large message blocks with CRC
- **Output:**
  - Block code with different length and breaks
1. Block coding

Example of Block Coding:

- Reed-Solomon Block:

  Viterbi-coded block + CRC-block

  188 bytes + 16 byte

- Recognition and correction of 8 error-bytes
2. Convolutional Coding

- Input:
  - Serial data stream with includes redundant information

- Output:
  - Decoded binary data stream
Use of the Viterbi Algorithm

1. Digital communication system

- Low Signal to noise ratio ratio
- Radio link
- Satellite connection
- Digital TV (MPEG-2/DVB)
- Geostationary satellite networks (VSAT)
- Speech recognition
- Magnetic recording
- Direct broadcast satellite systems (DBS)
2. Description of the Algorithm

- Generation of the data
- Convolutional encoder
- Parameters
- Channel symbols mapping
- Noise adding
- Channel symbol quantizing
- Viterbi Decoding
Generation of the data

- needed for simulation
- eg. using a random number generator
  - eg. rand(); in C
  - value less than half of the maximum value is a zero
  - any value greater or equal to half of the maximum value is a one
2. Description of the Algorithm

Convolutional encoder

INPUT
(k BITS/SEC)  FF  FF  Output

Diagram showing the convolutional encoder with input, fed through two FFs, and producing an output.
Convolutional encoder

State diagram

Encoder
Parameters

- eg. k=3
  - represent the code generator polynomials
- eg. m=2
  - number of shift-register
- eg. R=1/2
  - code rate (one inputbit $\rightarrow$ two outputbits)
Channel symbols mapping

- Is simply a matter of translating:
  - zeros to +1
  - ones to –1

- method is called living zero

This can be accomplished by performing the operation: \[ y = 1 - 2x \]
Noise adding

- generating Gaussian random numbers
- adding the scaled Gaussian random numbers to the channel symbols values
Channel symbol quantizing

- Hard-decision
- Soft-decision
Hard-decision

- quantized to one-bit precision
  - $< 0V = 1$
  - $> 0V = 0$
2. Description of the Algorithm

Soft-decision

- quantized with more than one bit of precision
- three or four bits of precision can perform about 2 dB better than hard-decision
- the usual quantization precision is three bits
3 Bit Quantizer
Viterbi Decoding

- Diagrams help to understand the algorithm
  - State diagram
  - Trellis diagram
2. Description of the Algorithm

State diagram / Trellis diagram
Viterbi decoding 1
Viterbi decoding 2
2. Description of the Algorithm

Trellis diagram

<table>
<thead>
<tr>
<th>current</th>
<th>next</th>
<th>sended bit</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>00</td>
<td>0</td>
</tr>
<tr>
<td>00</td>
<td>10</td>
<td>1</td>
</tr>
<tr>
<td>10</td>
<td>01</td>
<td>0</td>
</tr>
<tr>
<td>10</td>
<td>11</td>
<td>1</td>
</tr>
<tr>
<td>01</td>
<td>00</td>
<td>0</td>
</tr>
<tr>
<td>01</td>
<td>10</td>
<td>1</td>
</tr>
<tr>
<td>11</td>
<td>01</td>
<td>0</td>
</tr>
<tr>
<td>11</td>
<td>11</td>
<td>1</td>
</tr>
</tbody>
</table>
3. Advantages of the Viterbi Algorithm

- Self-Correction of the Code
- Minimization of the Transmitting Energy
- Minimization of the Bandwidth
- Other Advantages
Self-Correction of the Code

- very good ability to correct wrong transmitted bits ⇒ forward error correction
- done by adding of a redundant information
- state diagram offers a complete description of the system
Self-Correction of the Code

- State diagram
Minimization of the Transmitting Energy

- **BER**

Simulation Results for Rate 1/2 Convolutional Coding with Viterbi Decoding on an AWGN Channel with Various Convolutional Code Constraint Lengths

![Graph showing BER vs Eb/No (dB) for different code constraint lengths](image)
Minimization of the Transmitting Energy

- good designed satellite transmissions: BER < 10^{-6}
- professional transmissions: BER < 10^{-10}
- transmitting power: 4 Watt
Minimization of the Bandwidth

- possible to reconstruct lost data ⇒ used to save bandwidth
- more bandwidth needed, because of all the redundant information
- rate 1/2 ⇒ double of the bandwidth
- don’t send every third bit ⇒ able to reconstruct the dropped data
Minimization of the Bandwidth

- called puncturing of the code
- disadvantage: the transmitting energy have to been increased
Other Advantages

- fixed decoding time
- high transfer rate, up to 2 Mbps
- implementation in hardware and software
4. Presentation of the Software

- **task:**
  - constraint length: 7
  - code rate: 1/2
  - soft decision: 3 bit quantizer
Viterbi Algorithm

Thank You for Your Attention!