1. Leszek NOWOSIELSKI[1], 2. Bartosz DUDZIŃSKI[2], 3. Michał NOWOSIELSKI[3], 4. Aleksandra ŚLUBOWSKA[4]

Military University of Technology, Faculty of Electronics (1, 2), University Clinical Centre of the Medical University of Warsaw (3, 4)
ORCID: 1. 0000-0002-4230-4905; 2. 0000-0002-7404-0676

doi:10.15199/48.2023.07.26

Recognition of Convolutional Codes

Streszczenie. *Łączność radiowa pozostaje jak dotąd jedną z najskuteczniejszych i najbardziej rozpowszechnionych metod komunikacji. Dlatego potrzebne są narzędzia (takie jak Uniwersalny Demodulator Cyfrowy), które dawałyby odpowiednim służbom państwowym możliwość monitorowania nieznanych emisji radiowych. Niestety, producenci sprzętu i oprogramowania do automatycznego rozpoznawania protokołów komunikacji radiowej nie publikują stosowanych przez siebie rozwiązań, twierdząc, że są to informacje zastrzeżone. Autorzy artykułu nie odnaleźli w dostępnej literaturze wielu artykułów dotyczących automatycznego rozpoznawania kodów splotowych, stąd proponują taką metodę rozpoznawania kodu splotowego, która jest jednym z wielu elementów składowych systemu rozpoznawania protokołów komunikacji radiowej. Pierwsza część artykułu zawiera krótki opis procesu dekodowania kodu splotowego z wykorzystaniem algorytmu Viterbiego. Poniżej przedstawiono prezentację innowacyjnej metody automatycznego rozpoznawania typów kodów splotowych, opartej na monitorowaniu wartości metryk ścieżki podczas pracy dekodera Viterbiego. Na koniec przedstawiono wyniki pomiarów efektywności działania proponowanego algorytmu rozpoznawania kodu splotowego, uzyskane za pomocą symulacji komputerowej.*

Abstract*. Radio communication remains one of the most effective and prevalent methods of communication to date. Therefore, there is a need for tools (such as the Universal Digital Demodulator), which would give the appropriate state services the ability to monitor unknown radio emissions. Un-fortunately, manufacturers of hardware and software for automatic recognition of radio com-munication protocols do not publish the solutions they use, arguing that these solutions are proprietary information. The authors of the article did not find many articles on automatic recognition of convolutional codes in the available literature, and thus propose such a convolu-tional code recognition method, which is one of many components of the radio communication protocol recognition system. The first part of the article contains a short description of the convolutional code decoding process using the Viterbi algorithm. What follows is a presentation of an innovative method of automatic recognition of convolutional code types, based on monitoring of path metrics values during the operation of the Viterbi decoder. Finally, the results of operational efficiency measurements for the proposed algorithm convolutional code recognition algorithm, achieved using computer simulation, are presented. (****Rozpoznawanie kodów splotowych****).*

Słowa kluczowe: kody splotowe, FEC, dekoder Viterbiego
Keywords: convolutional codes, FEC, Viterbi decoder

Introduction

In certain military and civilian applications, there is a use for devices, which au-tomatically recognize, demodulate and decode signals transmitted by radio transmis-sion systems. Each radio transmission system is a unique set of distinct parameters. These include:

- RF signal parameters such as: frequency, modulation type, modulation parameters (e.g. radio frequency shift for FSK modulation), modulation rate;
- transmission protocol parameters such as: operating mode (synchro-nous, asynchronous), scrambler implementation, heading, address, synchronization sequence, end of message sequence etc;
- error correcting code types (convolutional codes, block codes);
- alphabets (e.g. ASCII alphabet).

From the perspective of an operator of an automatic radio transmission recogni-tion and decoding equipment, it is important that said equipment has the ability to automatically recognize a particular radio transmission system based on these pa-rameters. After proper recognition of the data transmission system, that equipment should then perform the process of decoding the information being sent, acting in ac-cordance with the rules specific for a recognized transmission system.

The information on convolutional codes presented in the current technical liter-ature applies to: punctured convolutional codes and combined coding [1,5,6,8], Viterbi decoder [2,4], forward error correction (FEC) [3].

Current literature [9, 10, 11] focuses on blind recognition of convolutional encod-er parameters based on correlation methods and which require calculation of syn-dromes. All methods presented in literature are time consumed process.

The method of automatic convolutional code recognition algorithm proposed in the paper, is basing on monitoring the increment value of the path metrics during the operation of the Viterbi decoder. In the case when the Viterbi decoder calculates paths metric increments based on the trellis diagram corresponding to the convolutional coder, which was used to generate the received sequence of code combinations at transmitter site, then the value of metrics increments corresponding to one of paths will be much greater than the value of increments corresponding to the remainder of paths, which survived during the operation of the decoder. In the case when the Viterbi decoder calculates increments of path metrics based on the trellis diagram that does not correspond to convolutional coder, which was used to generate the received sequence of code combinations, then the increments of metrics for all paths that sur-vived during the operation of the Viterbi decoder will have a small value.

Presented in the paper method of recognition of convolutional codes is imple-mented in the Universal Digital Demodulator (UDD), which was elaborated at the Military University of Technology in Poland, at the Faculty of Electronics. In the final version, UDD will automatically recognize, demodulate and decode radio transmis-sion signals, operating on specific system rules defined for the recognized transmis-sion system. The block diagram of the universal digital decoder is shown in Fig.1.

The radio transmission system recognition process includes a node for automatic recognition and decoding of convolutional codes used by the transmission system as forward correction codes (FEC). The node responsible for handling this task is high-lighted in Fig 1 with a shade of grey, and the method for automatic recognition of convolutional codes type is described in this article. In order to measure the effective-ness of the proposed method, extensive computer simulation was performed using custom software written in C++, and the results of this simulation are presented in the article as diagrams. To evaluate the effectiveness of our solution, we chose to measure the average increment value of the path metrics Δ assigned to l

branches of the coder trellis diagram, as well as the standard deviation of increment value for the path met-rics σ through the BSC (Binary Symmetric Channel). The conclusions worked out on the basis of the simulation results will aid in the implementation of the aforemen-tioned method in the final version of the Universal Digital Demodulator.

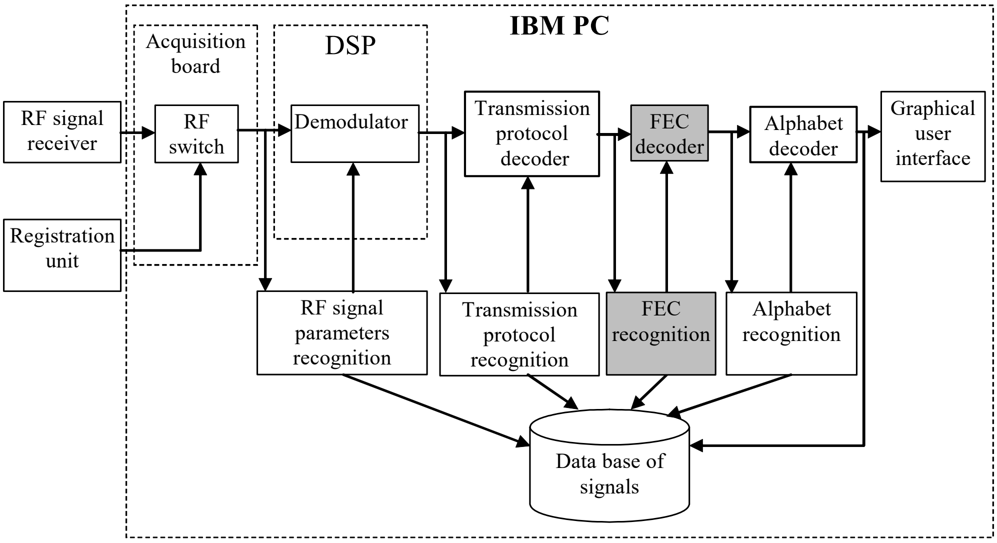


Figure 1. The Universal Digital Demodulator block diagram.

Convolutional Codes and Decoding Thereof

Integrated convolutional code coders and decoders are manufactured by such large international companies as QUALCOMM [2] or STANFORD TELECOM [13], which is a testament to the popularity of convolutional codes. These products are often used in error correction systems of radio communication equipment. In addition, convolutional codes have appeared in the military standard for HF modems [12], which suggests that this technique is now well -established in radio communications. A set of binary convolution-al codes with the minimum free distance dfree can be found in professional literature [8]. Therefore, it is purposeful to implement an automatic radio transmission protocol recognition and decoding algorithm, so that the radio transmission recognition and decoding equipment will be able to automatically recognize these codes without human interven-tion.

The optimal method for decoding convolutional codes that minimizes the probability of decoder decision error is to decode according to the rule of maximum probability, assuming that all messages are equally probable. This principle, in the case of the convolu-tional codes, is handled by the Viterbi algorithm. This algorithm executes the decoding process by selecting the path describing the operation of the coder in a trellis diagram, for which the coded sequence differs itself from the sequence received at the lowest number of places at the trellis' path. Fig. 2 shows an example convolutional coder block diagram and a trellis diagram. The algorithm is executed by calculating a metric for each possible path in the trellis diagram. The metric for a given path is defined by the Hamming distance between the code sequence determined by the given path and the sequence received. Therefore, for each node of the trellis diagram the algorithm compares paths incoming to the node. The path with the smallest metric is selected. The calculations are repeated for each level within the trellis diagram in the scope of the decoder memory. The paths selected by the algorithm are called "survived paths". During the operation, the decoder de-fines in the trellis diagram a number of paths corresponding to the number of coder states.

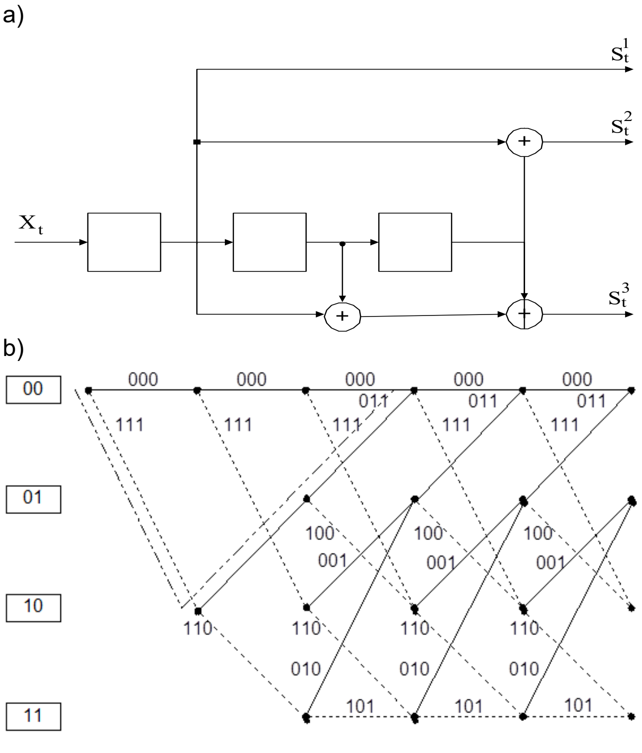


Figure 2. The example of convolutional coder block diagram (a) and trellis diagram (b) for code: m=3, k=1, n=3 and dfree=6.

The algorithm for automatic recognition of the convolutional codes

Correct operation of a Viterbi convolutional code decoder requires knowledge, on the receiving side, of the type of code used to encode the received sequence of code streams. Knowing the type of code, it is possible to define correctly the form of the trellis diagram being the base for the Viterbi decoder to decode the incoming stream of code combinations. The knowledge of the following coder parameters for convolutional codes is needed to de-fine the form of the trellis diagram:

- m-number of cells of the coder shift register;
- k-number of information bits incoming to the coder input in a unit of time;
- n-number of code bits received at the coder output in a unit of time;
- $\vec{g}_i$ -the coder connections vectors.

To ensure the correct operation of the Viterbi decoder, it is also necessary to achieve synchronization with the incoming stream of code sequences.

The proposed method for automatic code type recognition is based on monitoring the value of paths metric increments at the section of *l* in length of the trellis diagram branch-es, which survived during the operation of the Viterbi decoder. If we denote the metric value of the *j*-th path in the *i*-th and (*i-l*)-th step of decoding by $\gamma_i^{(j)}$ and $\gamma_{i-l}^{(j)}$ respec-tively, then the increment of path metric value related to the branch l is described by the formula (1).

$$\Delta_{l,i}^{(j)} = \gamma_i^{(j)} - \gamma_{i-l}^{(j)} \tag{1}$$

In the case when the Viterbi decoder calculates paths metric increments based on the trellis diagram corresponding to the convolutional coder, which was used to generate the received sequence of code combinations, then the value of metrics increments corre-sponding to one of paths will be much greater than the value of increments corresponding to the remainder of paths, which survived during the operation of the decoder. The value of the metric increment for the dominant path should be

$$\Delta_{l,i}^{(j)} = l \cdot n \tag{2}$$

where : l – monitored number of branches on the coder trellis diagram; n – number of coding bits related to one branch on the coder trellis diagram.

In the case when the Viterbi decoder calculates increments of path metrics based on the trellis diagram that does not correspond to convolutional coder, which was used to generate the received sequence of code combinations, then the increments of metrics for all paths that survived during the operation of the Viterbi decoder will have a small value of $\Delta_{l,i}^{(j)}$.

In the case when Viterbi decoder calculates increments of path metrics based on the trellis diagram that corresponds to convolutional coder, which was used to generate the received sequence of code combinations, and when the sequence of code combinations will be sent through a BSC discrete channel where independent errors are present with the probability of *p*, the situation described above will repeat itself, but with the difference that the value of the dominant path metric increment will be lower than the value defined by the dependence (2). The greater the probability of errors in the discrete channel, the greater the difference between these values.

The dependence of path metrics increments described above, calculated during the operation of the Viterbi decoder from the type of the trellis diagram (type of the code), which forms the basis for their calculation, can be used for automatic recognition of the convolutional code used for coding the received code combination sequences.

The procedure for detecting a convolutional code will show a match between a speci-fied trellis diagram (the type of convolutional code) and the received code combination sequences when $\Delta_{l,i} \geq t$ for one of the paths that survived, where *t* is the threshold value the quantity of which has been matched for a specified BSC discrete channel.

The value of the parameter *t* should be lower than *l·n*. The lower limit of the parame-ter *t* value depends on the value of error appearance probability in the discrete channel. The higher the error appearance probability value *p* in the discrete channel, the lower the value *t* should be.

The set of convolutional code types to be tested during the operation of the convolu-tional code recognition process can be limited to those codes, the coders and decoders of which are produced in the form of integrated circuits [2, 12, 13], as these codes are the most frequently used in practical applications.

A decoder of the convolutional code can correctly operate only with the assumption that the synchronization is operating correctly as well. Improper matching of the received bits with the ones assigned to suitable branches in the trellis diagram of the coder would cause a very large number of errors. To avoid this situation in practical applications, so-phisticated circuits are used to supervise and maintain the decoder's synchronization with the incoming stream of received symbols. A simple way to achieve synchronization is to monitor the incrementing value of path metrics, which survived during the operation of the Viterbi decoder. It is done in similarly to the recognition of the convolutional codes described above, the difference being that in this case, the form of the trellis diagram is not changed, whereas the sequence of code symbols received is periodically being shifted by one bit in relation to the bits corresponding to particular branches of the coder trellis dia-gram. After each such shift, the value of path metrics incrementing at the section of *l* in length for the branches of the trellis diagram that survived during the operation of the Viterbi decoder is checked. If the received sequence of code combinations is synchronized with the code combinations corresponding to particular branches of the coder trellis dia-gram, the increments value of metrics related to one of the paths will be much greater than the increments corresponding to other paths that survived during the operation of the de-coder. The metric increment value of the dominant path, in the case when an undisturbed code sequence is received, as well as in the case of when synchronization is achieved, is defined by the formula (2). The algorithm of the procedure of automatic recognition of convolutional codes described above is presented in Fig. 3.

Evaluation of the effectiveness of the convolutional code recognition procedure bycomputer simulation

In order to test the possibilities of automatic convolutional code recognition using the method of monitoring the increment metrics value of the paths, which survived during the operation of the Viterbi decoder, a computer simulation of the process was performed for a suitable code channel (Fig. 4). The statistical data obtained from that computer simulation confirmed that it is possible to automatically recognise the convolutional code type using this method.

In all simulations, the BSC independent error discrete channel model was used to generate errors arising in the discrete channel. The simulation program was written in C++.).

We have performed a series of computer simulations in order to test possibilities for recognizing a code type using the method of monitoring the increment metrics values of paths that survived during the operation of the Viterbi decoder. The resulting statistical data allowed us to evaluate the possibilities of detecting the type of the convolutional code.

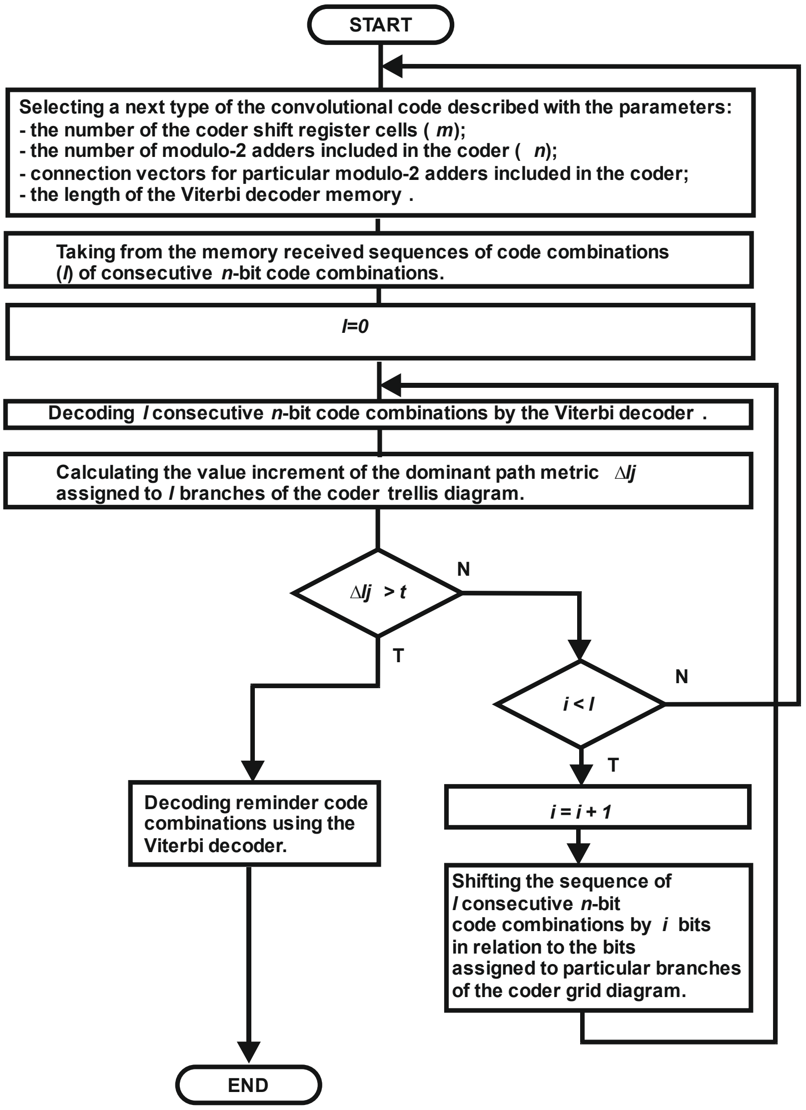


Figure 3. Automatic convolutional code recognition algorithm.

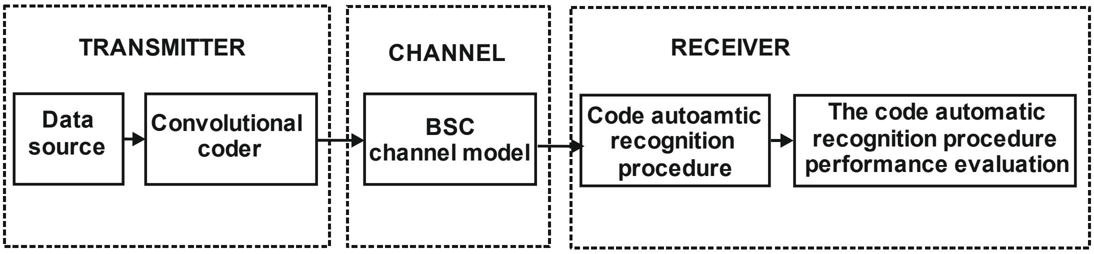


Figure 4. The computer simulation system block diagram.

The simulation tested the detecting possibilities for convolutional codes with a code rate of $R_c=1/2$ and $R_c=1/3$. For the detected code rate $R_c=1/2$, the code recommended by the military standard MIL STD 188 110C [12] was used with the parameters:

- number of shift register cells of the coder $m=7$;
- number of modulo-2 adders being a part of the coder $n=2$;
- connection vectors of the coder written in binary code $\mathbf{g_1}$ = [1101101], $\mathbf{g_2}$=[1001111].

For the detected code of the rate $R_c=1/3$, the industry standard code was used with the parameters:

- number of shift register cells of the coder $m=7$;
- number modulo-2 adders being a part of the coder $n=3$;
- connection vectors of the coder written in binary code $\mathbf{g_1}$ =[1101101], $\mathbf{g_1}$ = [1001111], $\mathbf{g_3}$ =[1010011].

Following the operational algorithm of the simulation program (Fig. 4), at the beginning of each simulation, the binary combination code of incoming data was coded using one of the codes mentioned above. Then, errors generated by the procedure simulating the operation of the BSC discrete channel were introduced to the code combination sequences obtained from the output of the coder. The disturbed code combination sequence was then processed according to the automatic convolutional code recognition procedure (Fig. 3).

The simulation was performed using the Monte Carlo method, and in the result the average increment value of the path metrics Δ assigned to l branches of the coder trellis diagram, as well as the standard deviation of increment value for the path metrics σ were achieved. Based off simulation results, the values of these parameters were achieved as the function of:

- type of the code tested (type of the code, for simplifying the formulation, was marked with a number of shift register cells of the coder *m*);
- number *l* of grid diagram branches used for evaluation of the path metrics increment value;
- error appearance probability in the BSC discrete channel *p*;
- information contents of the code.

Simulation results and their analyses

Fig. 5 shows the increment value of the path metric Δ and it standard deviation σ assigned to *l=5* branches of the coder trellis diagram as the function of code type tested, marked with a number of shift register cells of the coder *m*, for codes with a code rate of $R_c=1/3$. It can be seen from this diagram that when the code tested by the recognition procedure is compatible with the code used for encoding the received code sequence, then the value of parameter Δ has a much higher value than the one assigned for other tested convolutional codes. The lower error appearance probability in the discrete channel *p,* the higher is this difference. In the case of non-disturbed code sequence (*p=0*) the increment value of the path metric is $\Delta_{l,i}^{(j)}=l\cdot n=5\cdot 3=15$. Fig. 5 also presents the standard deviation of the path metric value.

For codes which are incompatible with the code used to encode the received code sequence, the value of the standard deviation is approximately constant.

If the codes are compatible with the tested code, the standard deviation value goes lower in the function of decreasing tge probability value of error appearance in the BSC discrete channel *p*.

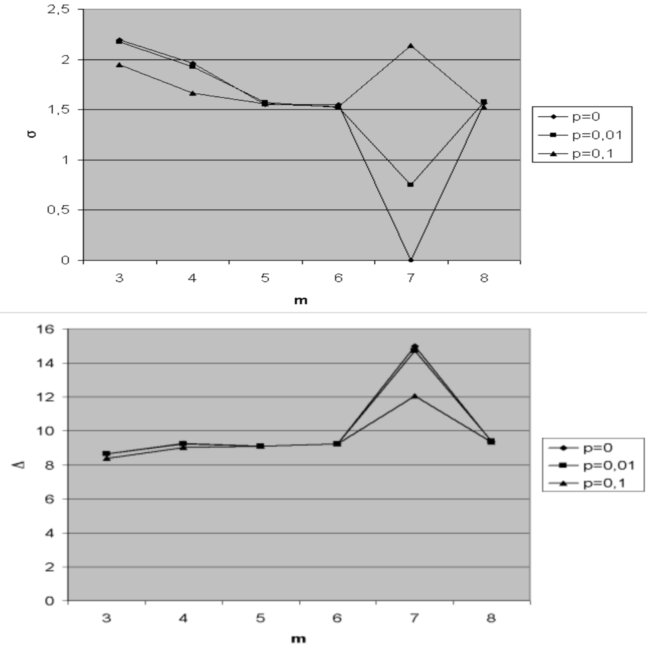


Figure 5. The increment value of the path metric Δ and its standard deviation σ assigned to l=5 branches of the coder trellis diagram as the function of the cell number of the coder shift register m, for codes with a code rate of Rc=1/3.

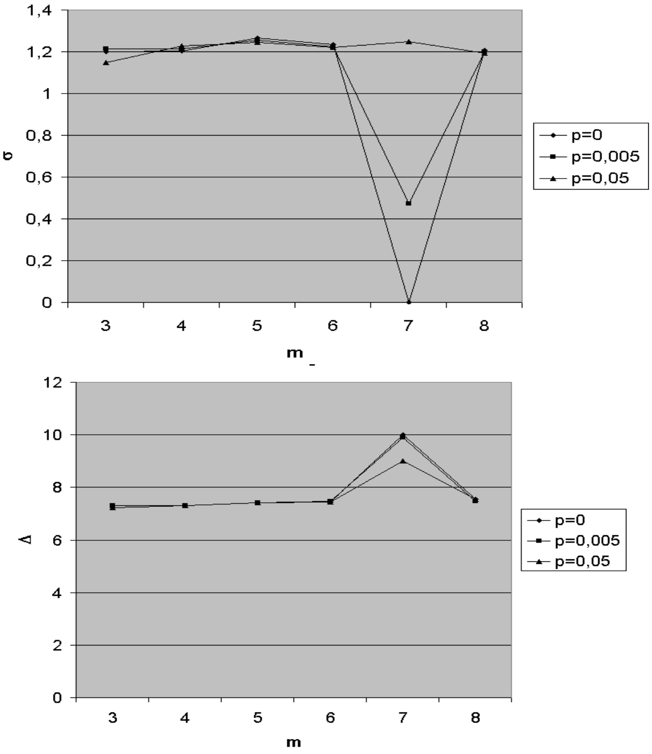


Figure 6. The increment value of the path metric Δ and its standard deviation σ assigned to l=5 branches of the coder trellis diagram as the function of the cell number of the coder shift register m, for codes with a code rate of Rc=1/2.

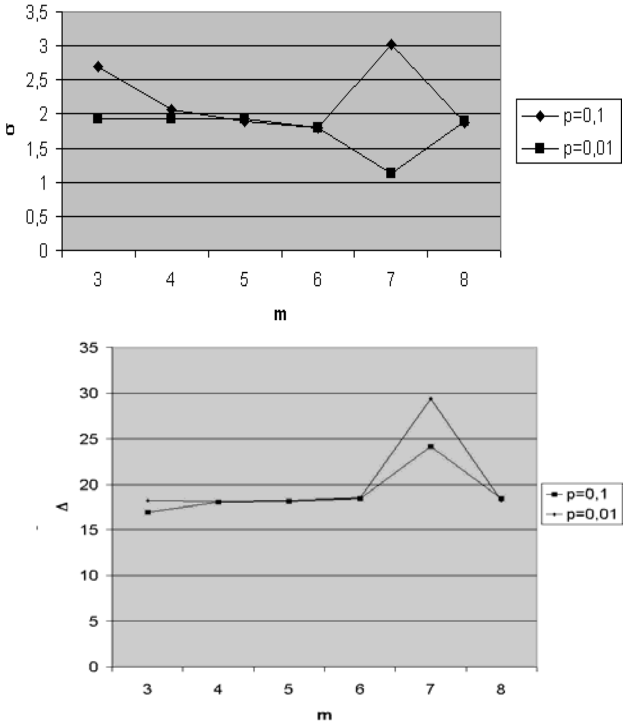
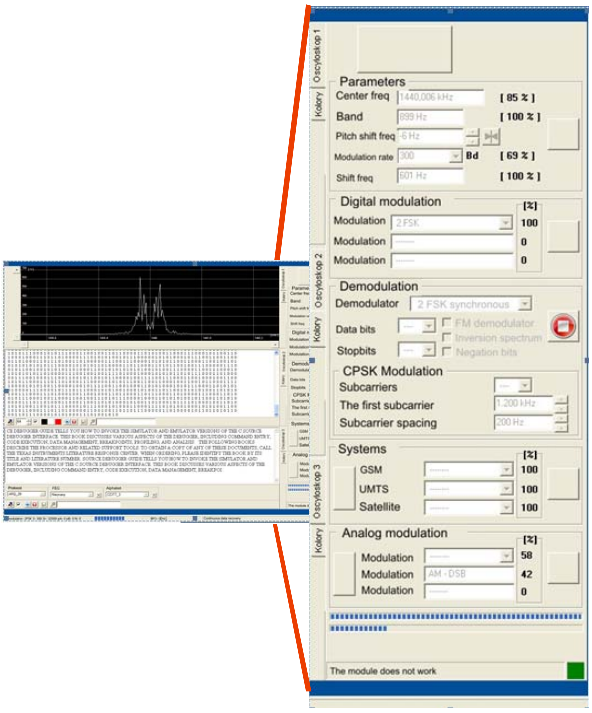


Figure 7. The increment value of the path metric Δ and its standard deviation σ assigned to l=10 branches of the coder trellis diagram as the function of the cell number of the coder shift register m, for codes with a code rate of Rc=1/3.

Fig. 6 shows the increment value of the path metric Δ and it standard deviation σ assigned to *l=5* branches of the coder grid diagram, as well as its standard deviation in the function of the number of the coder shift register cells *m*, for codes of information contents $R_c=1/2$. It can be seen from this diagram that, when the code tested by the recognition procedure is compatible with the code used for encoding the receiving code sequence, then the value of the parameter Δ has a much higher value than the one assigned for other convolutional codes tested. The maximum value of the parameter Δ for the code with the information contents $R_c=1/2$ and $l=5$ is $\Delta_{l,i}^{(j)}=l\cdot n$ $5\cdot 2=10$. When the value of parameter l increases, the increment value of the path metric increases along with it. This can be seen in Fig. 7.



Parameters
Center freq 1440,006 kHz [85 %]
Band 899 Hz [100 %]
Pitch shift freq -6 Hz
Modulation rate 300 Bd [69 %]
Shift freq 601 Hz [100 %]

Digital modulation [%]
Modulation 2 FSK 100
Modulation ------ 0
Modulation ------ 0

Demodulation
Demodulator 2 FSK synchronous
Data bits ---
Stopbits ---
☐ FM demodulator
☐ Inversion spectrum
☐ Negation bits

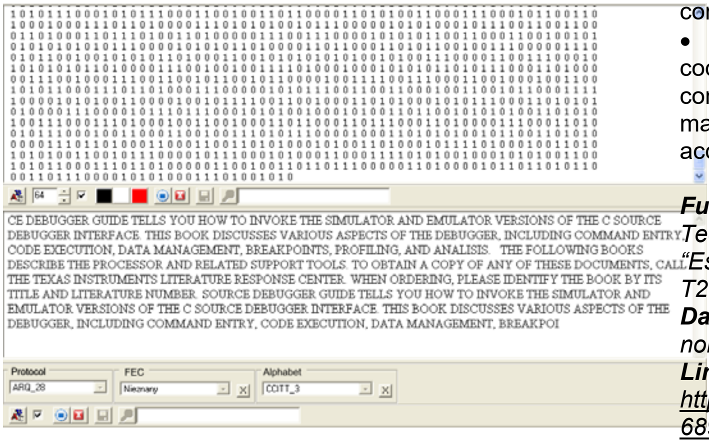
CPSK Modulation
Subcarriers ---
The first subcarrier 1.200 kHz
Subcarrier spacing 200 Hz

Systems [%]
GSM ------ 100
UMTS ------ 100
Satellite ------ 100

Analog modulation [%]
Modulation ------ 58
Modulation AM - DSB 42
Modulation ------ 0

The module does not work

Fig. 8. A view of the program window includes the decoded text and the transmission parameters.



```
1 0 1 0 1 1 1 0 0 0 1 0 1 0 1 1 1 0 0 0 1 1 0 0 1 0 0 1 1 0 1 1 0 0 0 0 1 1 0 1 0 1 0 0 1 1 0 0 0 1 1 1 0 0 0 1 0 1 1 0 0 1 1 0
1 0 0 0 1 0 0 0 1 1 1 0 1 1 0 1 0 0 0 0 1 1 1 0 1 0 1 0 1 0 0 1 0 1 1 1 0 0 0 0 0 1 0 1 0 1 0 0 0 1 0 1 1 1 0 0 1 1 0 0 1 1 0 0
0 1 1 0 1 0 0 0 1 1 0 1 1 1 0 1 0 0 1 1 0 1 0 0 0 0 0 1 1 1 0 0 1 1 1 0 0 0 0 1 0 1 0 1 0 1 1 0 0 1 1 0 0 0 1 1 0 0 1 0 0 1 0 1
0 1 0 1 0 1 0 1 0 1 0 1 1 1 0 0 0 0 1 0 1 0 1 0 1 1 0 1 0 0 0 0 1 1 0 1 1 0 0 0 1 1 0 1 0 1 1 0 0 1 0 0 1 1 1 0 0 0 0 0 1 1 1 0
0 1 0 1 1 0 0 1 0 0 1 0 1 0 1 0 1 1 0 1 0 0 0 1 1 0 0 1 0 1 0 1 0 1 0 1 0 1 0 0 1 0 1 0 0 1 1 1 0 0 0 0 1 1 0 0 1 1 1 0 0 0 1 0
1 0 1 0 1 0 1 0 1 1 0 1 0 0 0 0 1 1 1 0 0 1 0 0 1 0 0 1 1 1 1 0 1 0 0 0 1 0 0 0 1 0 1 0 1 0 1 1 0 1 0 1 1 1 0 0 0 1 1 0 1 0 0 0
0 0 1 1 1 0 0 1 0 0 0 1 1 1 0 0 1 1 0 0 1 0 1 1 0 0 1 0 1 1 0 0 0 0 1 0 0 1 1 1 1 0 0 1 1 1 0 0 0 1 1 0 0 1 0 0 0 1 0 0 1 1 0 0
1 0 1 0 1 1 0 0 0 1 1 1 0 1 1 0 1 0 0 0 1 1 1 0 0 0 0 0 1 0 1 0 1 1 1 0 0 1 1 0 0 1 1 0 0 0 1 0 0 1 1 0 0 1 0 1 1 0 0 0 1 1 1 1
1 0 0 0 0 1 0 1 0 1 0 0 1 1 0 0 0 0 1 0 0 1 0 1 1 1 1 0 0 1 1 0 0 1 0 0 0 1 1 0 1 0 1 0 0 0 1 0 1 0 1 1 1 0 0 0 1 1 0 1 0 1 0 1
0 1 0 0 0 0 1 1 1 0 0 0 0 1 0 1 1 1 0 1 1 1 0 0 0 1 0 1 0 1 0 0 1 0 0 0 1 0 1 0 0 1 1 0 1 1 0 0 1 0 1 0 1 0 1 0 0 1 1 0 1 0 1 0
1 0 0 1 1 1 0 0 0 1 1 1 0 1 0 0 0 1 0 0 1 1 0 0 1 0 0 0 1 1 0 1 1 0 0 0 1 1 0 1 1 1 0 0 0 1 1 0 1 0 0 0 0 1 1 1 0 0 0 1 1 0 1 0
0 1 0 1 1 1 0 0 0 1 0 0 1 1 0 0 0 1 1 0 0 1 0 0 1 1 1 0 1 0 1 1 1 0 0 0 0 1 0 0 0 1 1 0 1 0 1 0 1 0 0 1 0 1 1 0 0 1 1 0 1 0 1 0
0 0 0 0 1 1 1 0 1 1 0 1 0 0 0 1 1 0 0 0 1 1 0 0 1 1 0 1 0 1 0 0 0 1 0 1 0 0 1 1 0 1 0 0 0 1 0 1 1 1 1 0 1 0 0 0 1 0 0 1 1 0 1 0
1 0 1 0 1 0 0 1 1 0 0 1 0 1 1 1 0 0 0 0 1 0 1 1 1 0 0 0 1 0 1 0 0 0 1 1 0 0 0 1 1 1 1 0 1 0 1 0 0 1 0 0 0 1 0 1 0 1 0 0 1 1 0 0
1 0 1 0 1 1 0 0 0 1 1 1 0 1 1 0 1 0 0 0 0 0 1 1 0 0 1 0 0 1 1 0 1 1 0 1 1 1 0 0 0 0 0 1 1 0 1 0 0 0 0 1 0 1 1 0 1 1 0 1 0 1 1 0
0 0 1 1 0 1 1 1 0 0 0 0 1 0 1 0 1 0 0 0 1 1 1 0 1 0 0 1 0 1 0
```

CE DEBUGGER GUIDE TELLS YOU HOW TO INVOKE THE SIMULATOR AND EMULATOR VERSIONS OF THE C SOURCE DEBUGGER INTERFACE. THIS BOOK DISCUSSES VARIOUS ASPECTS OF THE DEBUGGER, INCLUDING COMMAND ENTRY, CODE EXECUTION, DATA MANAGEMENT, BREAKPOINTS, PROFILING, AND ANALISIS. THE FOLLOWING BOOKS DESCRIBE THE PROCESSOR AND RELATED SUPPORT TOOLS. TO OBTAIN A COPY OF ANY OF THESE DOCUMENTS, CALL THE TEXAS INSTRUMENTS LITERATURE RESPONSE CENTER. WHEN ORDERING, PLEASE IDENTIFY THE BOOK BY ITS TITLE AND LITERATURE NUMBER. SOURCE DEBUGGER GUIDE TELLS YOU HOW TO INVOKE THE SIMULATOR AND EMULATOR VERSIONS OF THE C SOURCE DEBUGGER INTERFACE. THIS BOOK DISCUSSES VARIOUS ASPECTS OF THE DEBUGGER, INCLUDING COMMAND ENTRY, CODE EXECUTION, DATA MANAGEMENT, BREAKPOI

Protocol ARQ_28
FEC Nieznany
Alphabet CCITT_3

Fig. 9. View of windows containing values of demodulated bits and decoded text of the message received.

Application of the proposed convolutional code recognition method

The convolutional code recognition algorithms described above have been implemented in the form of a software application, which executes the recognition process on a standard PC, as part of a project executed by the authors of this article. The block diagram of Universal Digital Decoder is presented in Fig. 1. The main window of the application is presented in Fig. 8. Fields related to the recognition of basic radio signal parameters are shown on the right side of the figure. These fields are related to the initial recognizing parameters necessary for the recognition of radio protocols, redundant convolutional codes, and alphabets.

Two program windows are visible in Fig. 9. The first window presents the values of demodulated bits – the results of demodulator operation. The second window shows the contents of message decoded after the recognition of radio protocol, redundant coding FEC and the alphabet.

Summary

The automatic convolutional code recognition algorithm proposed in the paper, based on monitoring the increment value of the path metrics during the operation of the Viterbi decoder has the following characteristics:

- the difference between increment path metric increases together with the number l of branches of the trellis diagram, which are used for evaluation of path metric increment for matched codes, and the path metric increment value for unmatched codes. For big values of the parameter l it is therefore easier to distinguish the matched code from an unmatched one;
- the delay introduced by the automatic recognition circuit of the convolutional codes increases together with increasing the value of the parameter l;
- the threshold value t of the path metric increment assigned to l branches of the coder trellis diagram used for detecting the code type should be lower than $l\cdot n$. The lower limit of the value for the parameter t depends on the errors appearance probability value in the discrete channel. The greater error appearance probability value in the discrete channel, the lower value t should be;
- in order to decrease time needed for recognizing the convolutional code the number of the convolutional codes tested should be limited to the necessary minimum (of codes met the most frequently in practice);
- for correct functioning the recognition of the convolutional codes procedure it is necessary to achieve after-element synchronization with the received code combination sequence;
- the presented recognition method of the convolutional codes can be used for reconditioning of the puncturing convolutional codes, of course, after taking the puncturing matrix used at the transmitting puncturing matrix into accoun

Funding: *This work was financed by Military University of Technology under research project UGB-22-859/2023 on “Estimation of electromagnetic compatibility between DVB-T2 and 4G/5G in the 700 MHz band”.*

Data Availability Statement: *This article is based on norms contained in the literature [1-7]*

Link:
https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6899998&tag=1

Conflicts of Interest: *The authors declare no conflicts of interest.*

REFERENCES

[1]. L.H. Charles Lee, Convolutional coding Fundamentals and applications, 1997.

[2]. QUALCOMM Incorporated, ASIC Products, *Forward Error Correction Products Data Book*, 80-24128 1

[3]. L. Nowosielski, M. Wnuk, *The performance of the algorithm for automatic FEC codes recognition*, 20th International Conference on Microwaves, Radar and Wireless

Communications, MIKON 2014, Gdansk, Poland, 16 June 2014 through 18 June 2014, ISBN: 978 839315252 0

[4]. A. A. Kurmukova, F. I. Ivanov & V. V. Zyablov, Theoretical and Experimental Upper and Lower Bounds on the Efficiency of Convolutional Codes in a Binary Symmetric Channel, Problems of Information Transmission volume 58, pages122–136, 2022.

[5]. Daniel J. Costello; Michael Lentmaier; David G. M. Mitchell, New perspectives on braided convolutional codes, 9th International Symposium on Turbo Codes and Iterative Information Processing (ISTC), ISSN: 2165-4719, 2016

[6]. Julia Lieb, Joachim Rosenthal, *Erasure decoding of convolutional codes using first-order representations*, Mathematics of Control, Signals, and Systems volume 33, pages499–513, 2021

[7]. Almeida PJ, Lieb J, *Complete j-MDP convolutional codes*. IEEE Trans Inf Theory 66(12):7348–7359, 2020

[8]. Ajay Dholakia, Introduction to Convolutional Codes with Applications, 1994

[9]. Shaojing Su, Jing Zhou, Zhiping Huang, Chunwu Liu, Yimeng Zhang, *Blind identification of convolutional encoder parameters,* ScientificWorldJournal. 2014;2014:798612. doi: 10.1155/2014/798612. Epub 2014 May 21

[10]. Shaojing Su, Jing Zhou, Zhiping Huang, Chunwu Liu, and Yimeng Zhang, *Blind Identification of Convolutional Encoder Parameters,*e Scientific World Journal Volume 2014, Article ID 798612

[11]. Wang Yao, Wang Feng-hua, Huang Zhi-tao, Blind Recognition of (n, k, m) *Convolutional Code Based on Local Decision in a Noisy Environment,* International Conference on Automation, Mechanical Control and Computational Engineering (AMCCE 2015)

[12]. MILD-STD-188-100C, Deparment of Defense Interface Standard, *Interoperability and performance standards for data modems*, 23 Septermber 2011

[13]. https://www.datasheetarchive.com/Stanford%20Telecom-datasheet.html