Simulation Results of Interactive Learning for Encoders and Decoders

Adriana Borodzhieva[1], Galia Marinova[2] and Tzvetomir Vassilev[3]

[1]Adriana Borodzhieva is with the Faculty of Electrical Engineering
Electronics and Automation at Ruse University
8 Studentska Str., Ruse 7017, Bulgaria
aborodjieva@ecs.uniruse.bg

[2]Galia Marinova is with the Telecommunications Faculty
Technical University of Sofia, 8 Kl. Ohridski Blvd., Sofia 1000
Bulgaria
gim@tu-sofia.bg

[3]Tzvetomir Vassilev is with the Faculty of Natural Sciences and Education at Ruse University
8 Studentska Str., Ruse 7017
Bulgaria
TVassilev@uni-ruse.bg

ABSTRACT: *We in this work have presented the simulation results of the convolutional encoders and decoders. With the use of MATLAB, we have presented an interactive learning module with graphical user interface. We have presented the framework for the functionality and features of the interactive learning module. When teaching the security of the telecommunication systems, we use the interactive learning module. This part adds to curriculum of the telecommunication system security.*

Copyright: with Author

1. Introduction

Numerous existing interactive program modules only allow encoding and decoding of binary sequences using convolutional encoders/decoders [9, 10], but they do not keep a database of all valid combinations of polynomial generators and the corresponding value of the parameter free distance, and a database of the best convolutional encoders, the encoders with maximum free distance defining the corrective capabilities of the encoders.

For example, the e-learning interactive course "Channel coding and decoding" [9] is organized in 3 parts: lecturebased course,

exercises and dynamic simulations. The main objectives of this tool are: explaining through illustrations the realization of the encoder, the decoding process, error detection and correction, performances of the codes.

Based on authors work on convolutional encoders during the last several years, an Interactive Learning Module, which integrates processes "divide" and "search" to find candidates for the best encoders and appropriate learning content, is developed in order to assist students in learning the topic of convolutional encoders analysis. The Interactive Learning Module described in the paper is a part of a software system for convolutional encoding/decoding. It is based on a set of MATLAB tools developed and tested previously [6, 7, 8]. A special Graphical User Interface (GUI) is realized to facilitate interactivity and learning process.

The learning content in the Interactive Learning Module covers subjects from communication circuit architecture to illustrations on the choice of connections between the adders and the register's flip-flops in the convolutional encoder's structure influencing the characteristics of the code, and changes in the choice of connections creating different codes.

The connections are not selected and changed randomly. The problem of selecting connections giving optimal distance properties is complex and generally unsolvable, but for any value of the constraint length of the code less than 20, the best codes are found [1, 2, 3, 4, 5]. The Interactive Learning Module is intended for simulation study of convolutional encoders and decoders in presence of noise.

The Interactive Learning Module is implemented using MATLAB and GUIDE (Graphical User Interface Development Environment). First the software system for convolutional encoding is described briefly, then the GUI developed and finally the Interactive Learning Module functionality.

2. Software System For Convolutional Encoding And Interactive Learning Module Implementing "Divide And Search" Procedure

2.1. Software system for convolutional encoding

Using the computational and graphical environment MATLAB and its extensions Communications Toolbox and Symbolic Math Toolboxes [11] a software system for simulation study of convolutional encoders and decoders in the presence of noise is implemented. The system integrates four modules: Module 1: "Divide and search", Module 2:

"Convolutional encoding and decoding", Module 3:

"Determining the performance of convolutional encoders", and Module 4: "Simulation study of cascaded convolutional encoders" (Fig. 1).
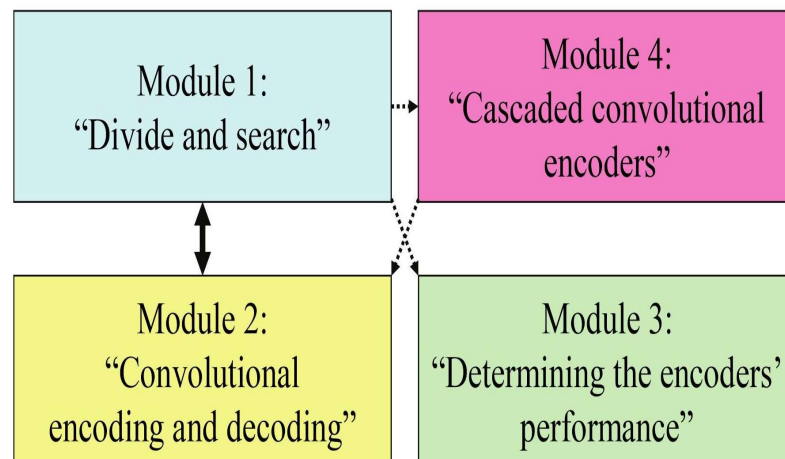

Figure 1. Architecture of the system implemented and connectivity of the four modules of the system

The first module performs "Divide and search" procedure involving the processes "dividing" the combinations of generator polynomials into valid and non-valid, and "searching" the valid combinations to find the encoders, candidates for the best encoders.

2.2. Layout and Functionality of GUI

The layout of the Interactive Learning Module with GUI is presented in Fig. 2. The layout is organized in 5 panels:

• **Panel 1:** Convolutional encoders;

• **Panel 2:** 4 push-buttons giving information about Module 1;

• **Panel 3:** Parameters of the studied group of convolutional encoders;

• **Panel 4:** Run;

• **Panel 5:** Visualization of the databases.

These five panels in the layout with the push-buttons and pop-up menus are described in details further.

Pressing the "Info" push-button (Fig. 2, panel 2) the architecture of the system implemented is shown in a separate graphics window, as the connection of the four modules of the system are given, as well as information about the use of Module 1: "Divide and search" (Fig. 1).

Below a "Convolutional encoders" panel (Fig. 2, panel 1) with two push-buttons and two pop-up menus is located. It supports learning content on the topics of digital communication systems and convolutional encoders.

The "Convolutional encoders" panel (Fig. 2, panel 1) allows students to learn some basic concepts in the theory of convolutional codes. When pushing the first push-button ("What is DCS?") the structure of the functional diagram of a digital
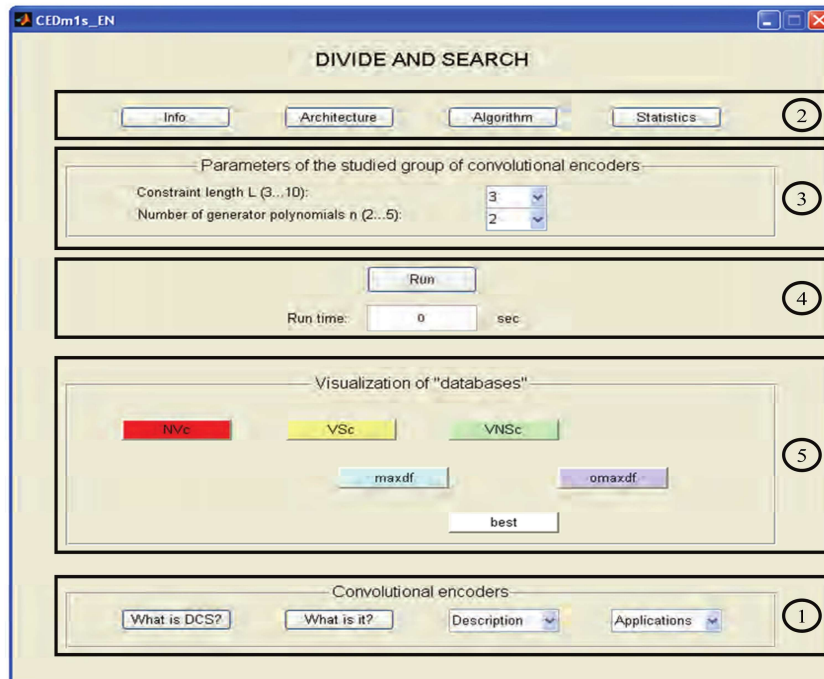

Figure 2. Layout of the Interactive Learning Module with GUI

communication system (DCS) is shown in a separate graphics window where the blocks "channel coding" (in the transmitter) and "channel decoding" (in the receiver) in a digital communication system are marked in blue, and they are subject of the study using the implemented application.

When pushing the second push-button ("What is it?") the structure of a convolutional encoder and a brief explanation of its principle of operation are shown in a graphical window.

From the "Description" pop-up menu a representation model (connections vectors, polynomial generators, state diagram, tree diagram, trellis diagram) of a convolutional encoder is chosen from six possible options. The model selected is displayed in a separate graphical window. Figure 3 illustrates the selection of connections vectors option) and information relevant to the representation model of the encoder for the simplest convolutional encoder with constraint length $L = 3$ and number of generator polynomials $n = 2$, with generators 111 and 101, respectively is given in details.

From the "Applications" pop-up menu (Fig. 2, panel 1) one of seven proposed applications of the convolutional encoders in the field of telecommunications is chosen. It aims to inspire students on the subject studied. For each of the selected options for learning content on convolutional encoders' applications in telecommunications (coding in a channel with error grouping, channel coding in wireless communications, encoding in control channels in GSM, encoding in control channels in IS-54, parameters of encoders, used in GSM and IS-95, history of space exploration, global satellite systems for personal communications) is displayed in a separate graphics window.
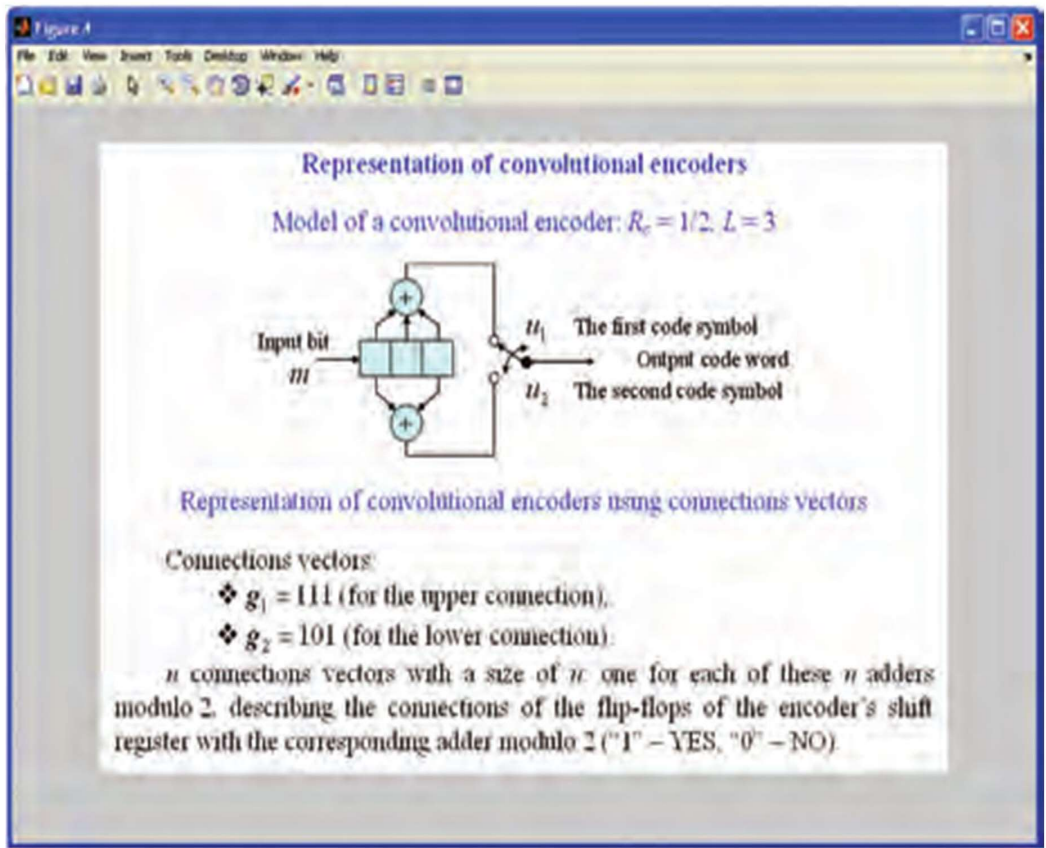

Figure 3. Learning content on the representation model of a convolutional encoder using connections vectors

2.3. Functionality of the Interactive Learning Module
Pressing the "Architecture" push-button the architecture of Module 1: "Divide and search" is displayed in a separate graphics window (Fig. 4), the third improved version [7].
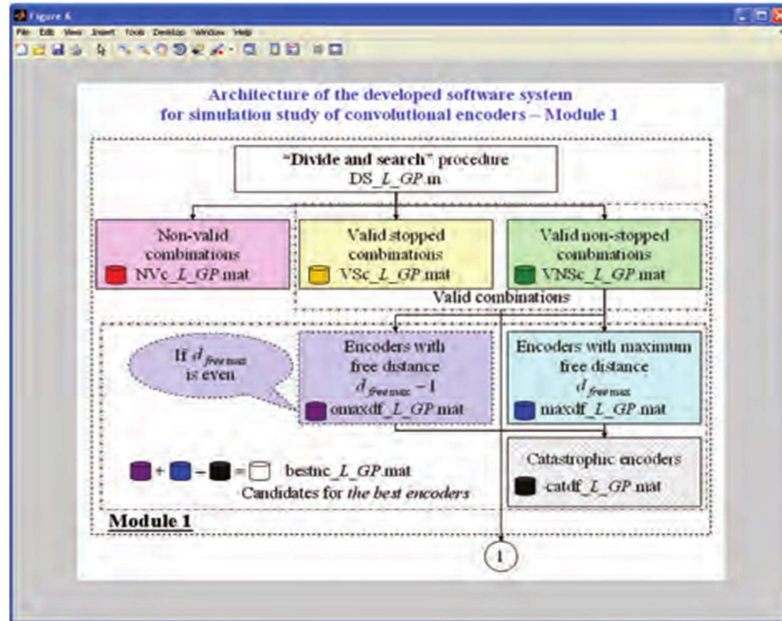
Figure 4. Architecture of Module 1: "Divide and search" when pressing the "Architecture" push-button The blocks of the architecture in Figure 4 are described below

After passing through all possible combinations of polynomial generators, they are divided into two groups:

1) non-valid combinations, stored in the database NVc_L_GP.mat and not examined;

2) valid combinations divided into two subgroups: valid stopped combinations, stored in the database VSc_L_GP.mat, for which the simulation is terminated after a fixed time (100 s) in an attempt to determine their free distance and valid non-stopped combinations, stored in the database VNSc_L_GP.mat, for which the simulation of the "divide and search" procedure has completed successfully, as their free distance is determined. From all valid non-stopped combinations in a separate database maxdf_L_GP.mat the combinations with a maximum value of the parameter free distance $d_{free\,max}$ are stored. If $d_{free\,max}$ is even the database omaxdf_L_GP.mat is generated where the encoders with free distance $d_{free\,max}$ -1 are kept. These two databases form the database best_L_GP.mat, which stores all possible candidates for the best encoders. The databases VNSc_L_GP.mat, maxdf_L_GP.mat, omaxdf_L_GP.mat and best_L_GP.mat are modified respectively into the databases VNSc_L_GP_df.mat, maxdf_L_GP_df.mat, omaxdf_L_GP_df.mat and best_L_GP_df.mat, containing information about the corresponding value of the parameter *free distance*.

After testing the candidates for the best encoders using Module 2 of the system all catastrophic encoders that allow catastrophic spreading of errors during the decoding process are stored in the database catdf_L_GP.mat. After eliminating the catastrophic encoders from the database best_L_GP.mat the database bestnc_L_GP.mat containing the encoders that actually have the chance to be the best is formed [7].

Module 1 was tested for the following cases: $L = 3$ and $GP = 2 \div 5$, $L = 4$ and $GP = 2 \div 5$, $L = 5$ and $GP = 2 \div 5$, $L = 6$ and $GP = 2 \div 3$, $L = 7$ and $GP = 2 \div 3$, $L = 8$ and $GP = 2 \div 3$, $L = 9$ and $GP = 2$, $L = 10$ and $GP = 2$. Using the mathematical models developed for cascaded convolutional encoders [8], "Divide and search" procedure is implemented for convolutional encoders with two registers and three generator polynomials used as internal encoder in the structure of the cascaded convolutional encoder, for the cases $L = [2\ 2]$ and $GP = 3$, $L = [3\ 3]$ and $GP = 3$.

The results of the simulation study are summarized in the table that is displayed in a separate graphics window when clicking on the "Statistics" push-button (Figure 5).

Number of combinations in different databases and maximum value of the parameter *free distance* for convolutional encoders for known constraint length L and number of generator polynomials GP

L GP	N	N_{NV}	N_{VS}	N_{VNS}	N_{maxdf}	N_{omaxdf}	N_{best}	N_{cdiff}	N_{home}	max df
3 2	49	17	4	28	2	0	2	8	24	5
3 3	343	53	8	282	3	15	18	6	12	8
3 4	2401	161	16	2224	34	112	146	0	146	10
3 5	16807	485	32	16290	10	0	10	0	10	13
4 2	225	89	14	122	16	32	48	6	42	6
4 3	3375	659	52	2664	12	54	66	0	66	10
4 4	50625	4721	170	45734	24	0	24	0	24	13
4 5	759375	33371	484	725520	830	3340	4170	0	4170	16
5 2	961	401	50	510	2	14	16	4	12	8
5 3	29791	6407	245	23139	6	72	78	0	78	12
5 4	923521	95849	1506	823166	54	528	582	6	576	16
5 5	28629151	1501943	7253	27119955	240	2970	3210	0	3210	20
6 2	3969	1697	176	2096	202	334	536	94	442	8
6 3	250047	56207	1366	192474	168	0	168	0	168	13
7 2	16129	6977	584	8568	14	182	196	24	172	10
7 3	2048383	470303	7235	1570845	216	0	216	12	204	15
8 2	65025	28289	1914	34822	1706	3568	5274	1118	4156	10
8 3	16581375	3846719	40645	12694011	49980	152709	202689	0	202689	16
9 2	261121	113921	6076	141124	138	1104	1242	118	1124	12
10 2	1046529	457217	19264	570048	13492	31838	45330	0	45330	12
2;2 3	729	129	138	462	174	0	174	600	0	3
3;3 3	117649	33839	16832	66978	2268	0	2268	2268	0	5

Figure 5. Analysis of convolutional encoders when pressing the "Statistics" push-button

In the "Parameters of the studied group of convolutional encoders" panel the user is able to choose from a pop-up menu the value of the constraint length L (3 to 10) and the number of generator polynomials n (2 to 5) of the encoder. These lists of options could be extended to other possible values. Pressing the "Run" push-button (Fig. 2, panel 4) a dialog box "Selection?" is shown (Fig. 6) with the following message: "The use of embedded databases is recommended, since in some cases the simulation takes too long time! Doyou want to follow the procedure "Divide and search?"".

If the user selects "No", the data from running "Divide and search" procedure embedded in the developed application is loaded and the user can use it "for granted" when working with other system's modules. If the user wants to continue to test the performance of the procedure he/she may need to be armed with great patience. The run time for the procedure is shown in seconds, when performing the "Divide and search" procedure. Otherwise the corresponding field is blank. For example, the simulation time for testing the group of convolutional encoders with L = 3, GP = 2 is 417.5930 sec.
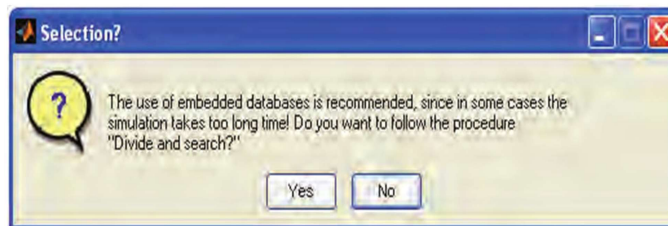

Figure 6. Dialog box "Selection?" when pressing the "Run" push-button

In the "Visualization of databases" button-group (Fig. 2, panel 5) there are six push-buttons: NVc, VSc, VNSc, maxdf, omaxdf and best. When pressing one of them the diagram of the algorithm implemented in the script DS_L_GP.m (only when L = 3, GP = 2) is given in a graphical window (Fig. 7) and the content of the corresponding database is visualized in the Variable Editor (Fig. 8) where the first two columns are the decimal generators (Fig. 8, block 1), and the third column (if available) is the value of the free distance (Fig. 8, block 2). The best encoders with maximum free distance are shown in Fig. 8, block 3.
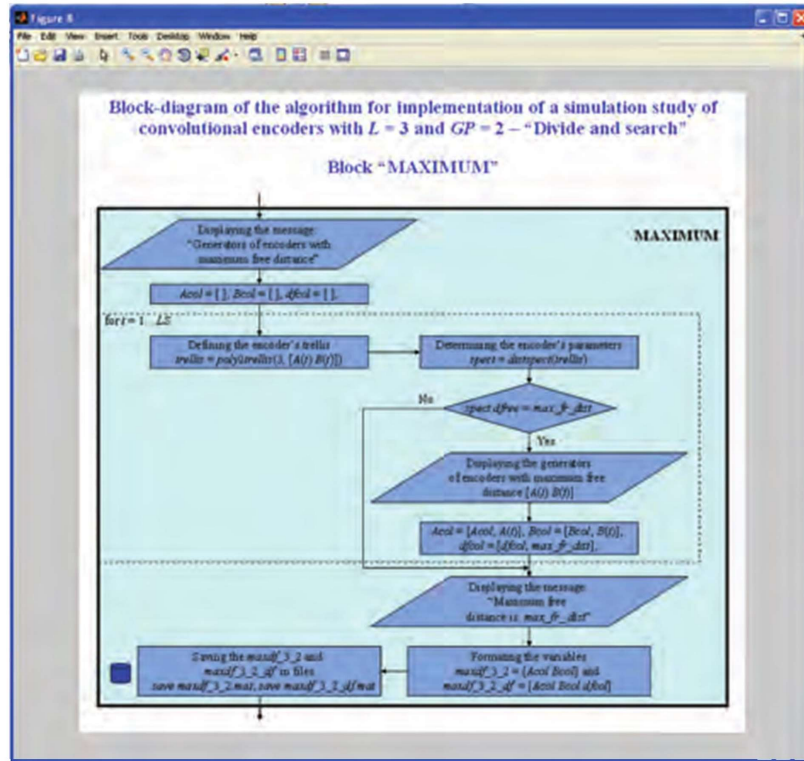
Figure 7. Block-diagram of the algorithm implemented in "Divide and search" procedure – block "Maximum" when pressing the "maxdf" push-button
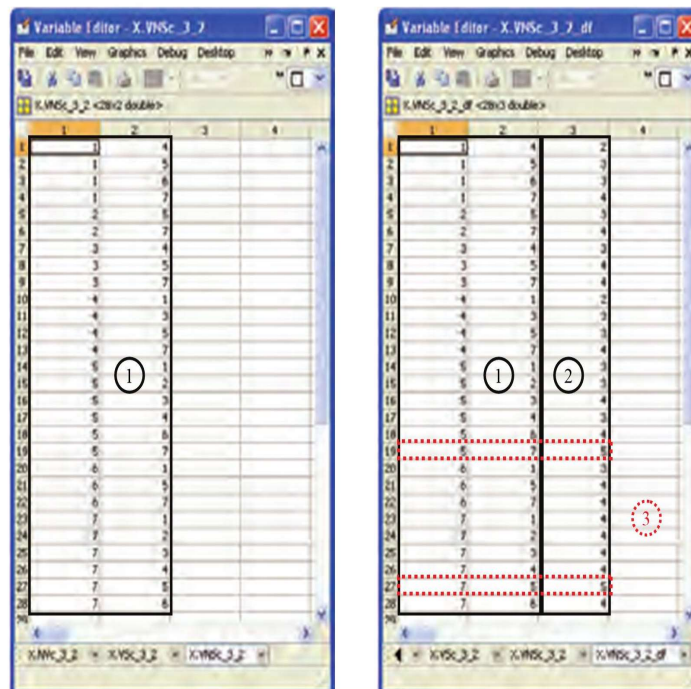

Figure 8. Variable Editor for the variables VNSc_3_2 and VNSc_3_2_df when pressing the "VNSc" push-button

3. Conclusion

The Interactive Learning Module will be used in teaching the course "Coding in Telecommunications Systems", included as optional in the curriculum of the specialty "Telecommunication Systems" for the Bachelor degree. Future work plans to develop similar applications covering other codes studied in this course: linear, cyclic, BCH-codes, Reed-Solomon codes and turbo-codes.

References

[1] Blahut, R.E. (1983). *Theory and Practice of Error Control Codes*. Addison-Wesley Publishing, Co.: Reading, USA.

[2] Curry, S.J. (1971). *Selection of Convolutional Codes Having Large Free Distance* [PhD Dissertation]. University of California, Los Angeles: USA.

[3] Larsen, K.J. (1973) Short convolutional codes with maximal free distance for rates 1/2, 1/3, and 1/4 (Corresp.). *IEEE Transactions on Information Theory*, 19, 371–372 [DOI: 10.1109/TIT.1973.1055014].

[4] Odenwalder, J.P. (1970). *Optimal Decoding of Convolutional Codes* [PhD Dissertation]. University of California, Los Angeles: USA.

[5] Sklar, B. (1988). *Digital Communications: Fundamentals and Applications*. Prentice Hall: NJ, USA.

[6] Borodzhieva, A. Software instruments for investigating convolutional encoding and decoding processes applied in communication systems, 14th edn, International Symposium for Design and Technology of Electronic Packaging, SIITME 2008. Predeal: Romania [Conference proceedings], pp. 135–139.

[7] Borodzhieva, A. *"Software System for Convolutional Encoders Investigation Using MATLAB and Communication Toolbox",Advanced Aspects of Theoretical Electrical Engineering, Sozopol '09, Regular Papers, Part 2*, pp. 117–123.

[8] Borodzhieva, A. *Simulation Investigation of Cascaded Convolutional Encoders Using MATLAB and Communication Toolbox. Advanced Aspects of Theoretical Electrical Engineering, Sozopol '09, Regular Papers, Part 2*, pp. 108–116.

[9] Nouvel, F. (2004) Channel coding and decoding: An e-learning interactive course. In: *Proceedings of the of international conference on Information and Communication Technologies: From Theory to Applications*, pp. 113–117.

[10] Tervo, R. Digital communications. www.ee.unb.ca/cgibin/tervo/viterbi.pl.

[11] *MathWorks Website*. www.mathworks.com.