AN ANALOG AND DIGITAL COMMUNICATIONS LABORATORY WITH MATLAB

Jan Chojcan[1], *Jacek Izydorczyk*[2] *and Grzegorz Dziwoki*[3]

Abstract — *Faculty of Automatic Control, Electronics and Computer Science of the Silesian University of Technology, in Gliwice Poland is developing Analog and Digital Communications Laboratory for undergraduate and graduate courses in Matlab. The main objective is to show a lot of modern communications technologies in virtual laboratory. Solution is a student book accompanied by Matlab's m-files and WWW pages using WWW extension of Matlab. The last is the core of the virtual laboratory. Exercises start from introduction to digital signal processing and go from the analog modulations techniques to the digital ones, digital equalization, digital synchronization, PLL technology, noise analysis, analog to digital conversion and speech compression. An example of the laboratory exercise on blind channels equalizations will be shown in details.*

Index Terms — *communication technology, channel equalization, distance learning, e-education, simulation, virtual laboratory*

INTRODUCTION

The continuous and fast development of the modern communications technologies is the serious challenge for the university centres if they want to educate students in this area effectively. There is a necessity to work out new teaching methods for the courses in the communications topics, mainly due to the high complexity of the modern communication problems. The best way to understand them is the laboratory exercises, which can give deep insight into the considered problems. How could such laboratory be set up? If we take into consideration such factors as for example the suggested topic (see below), exercises flexibility, number of students, a time limitations and in the end the distance learning issues, the software communications laboratory seems to be only reasonable approach.

Creation of such laboratory using the software equipment is being put into effect in the Institute of Electronics within the confines of The Fundamentals of Analog and Digital Communications course. The whole project is being realized under following assumptions:

- Consideration of wide scope of communication topics, including the basics of signal processing, analog and digital communications basics problems, and sophisticated and modern communication solutions. This large collection of topics is motivated by undergraduate and graduate courses.
- The simplicity and the clarity of the exercises without any loss of quality. Software environment allows combining different exercises in one form. It makes easier to deal with exercises. It has great significance if to take different level of students' skills into account (especially on graduate courses). It allows students to concentrate on communication algorithms, and how it works, without concern about any implementation issues. Additionally, the simulations approach is a convenient way of analysing the contemporary communication solutions, especially the latest digital technologies.
- Using the Matlab simulation environment. It provides core mathematics and graphical tools for algorithm analysis and development. Additionally, high specialized Matlab built-in functions (Toolboxes) support analysis.
- Unlimited access to the laboratory via Internet – a virtual laboratory. Matlab Web Server provides the ability to use Matlab via the World Wide Web. As a result, Matlab need not be running on the client machine and the end user has access to Matlab using only Netscape or Microsoft Web browser that makes distance learning possible. The WWW page preparation is required for every exercise.
- On-line tests and marks using special prepared web sites.
- Permanent development of the virtual laboratory. New exercises and modifications of old ones will be provided as quickly as possible or needed.

An example presented below is only a trial version of one of many laboratory exercises accessed via Web browser. On-line version will be available in the near future.

OVERVIEW OF THE LABORATORY EXERCISES

The laboratory exercises contain the wide scope of the communications problem. Although all problems are strictly combined between themselves, they could be divided into three thematic parts: signal processing techniques, the basics of analog and digital communications, advanced techniques in communications [1],[2].

[1] Jan Chojcan, Silesian University of Technology, Gliwice POLAND, passed away on June 8th
[2] Jacek Izydorczyk, Silesian University of Technology, Gliwice POLAND, izi@alfa.iele.polsl.gliwice.pl
[3] Grzegorz Dziwoki, Silesian University of Technology, Gliwice POLAND, dziwny@alfa.iele.polsl.gliwice.pl

Techniques of Signal Processing

This part of the laboratory exercises concerns the fundamentals of digital signal processing. They seem to be important exercises, because give essential set of knowledge and skills for further better exploration of the communication problems. Here is considered such problems as:

- Sampling, quantization and coding of analog signals – Pulse Code Modulation (PCM)
- Reconstruction of signals from samples
- Windowing of signals
- Signals convolution
- DFT transform and FFT algorithm
- Z-transform, difference equations
- Analysis of linear time-invariant systems in the z-domain
- Hilbert transform, frequency shifting
- Stochastic signal analysis
- DTMF signalling and Goertzel algorithm

Students also can gain skills at the digital filter design (FIR and IIR filters)

The Basics of Analog and Digital Communication

This part covers the basic problems of communication techniques. Following problems can be distinguished:

- Amplitude modulation – conventional AM, double-sideband AM (DSB), single-sideband AM (SSB)
- Frequency and phase modulation
- Analysis of analog demodulation techniques in the presence and absence of additive noise
- Digital modulation – Pulse Amplitude Modulation (PAM), Quadrature Amplitude Modulation (QAM), Phase Shift Keying (PSK), Frequency Shift Keying (FSK), differential coding of symbols
- Intersymbol interferences, Nyquist criterion, shaping filters
- Coherent and noncoherent detection
- Binary symmetric channel and additive white Gaussian noise channel – probability of error, channel capacity
- Error control coding – block and convolutional codes, Viterbi algorithm
- Synchronization – carrier and timing recovery

This set of problems is intended for undergraduate communication course. It will give the basic engineering knowledge, particularly if signal processing topics are taken into consideration too.

Advanced Techniques in Communication

The last part covers more sophisticated and the latest communication problems as for example:

- Adaptive filters with Least Mean Square (LMS) and Recursive Least Squares (RLS) adaptation algorithms
- Echo cancellation
- Linear channel equalization, decision feedback equalization (DFE) – training sequence algorithm, decision-directed algorithm
- Blind channel equalization methods
- Direct sequence spread spectrum systems, generation of pseudonoise sequences
- Speech analysis and synthesis, linear predictive coding analysis (LPC)
- Spectrum analysis of digitally modulated signals
- Modulation in wireless communication – Gaussian Minimum Shift Keying (GMSK) modulation
- Multicarrier systems, Orthogonal Frequency Division Multiplexing (OFDM) modulations, cyclic prefix and synchronization issues
- Adaptive Differential Pulse Code Modulation (ADPCM)
- Sigma-Delta ADC and DAC converters

These topics create large set of exercises that can be freely chosen for graduated courses. This course assumes the knowledge about communications and processing techniques at the level that can be obtained during undergraduate course.

Organizational Issues

The entire number of previously mentioned problems enable to make laboratory schedules not only for communications courses but also for closely related others as for example the signal processing course or the adaptive filters applications course. Next, different level of difficulties among suggested experiments satisfy the need of education on different level of student's skills. These two above mentioned conditions make the communication laboratory flexible.

The laboratory paper instruction and set of functions and scripts in Matlab code (m-files) are prepared for each laboratory exercise. M-files, which simplify developing considered problems, will be accessed via Internet. Each instruction is divided into three parts: extensive introduction to problem, the exercises before the Matlab experiments and Matlab experiments. Why is Matlab used? It is the one of the most universal environment for technical computing. It allows students to solve given problems in convenient programming way. Moreover, students can make use of experience achieved in Matlab programming when they make some hardware applications based on a signal processor. The rules of programming are similar in both cases.

The presented scope of communication and processing problems is open. It is obvious that frequent update of exercises is impossible for paper instructions. But instructions can take the form of electronic documents, which are published using on-line accessed and freely updated World Wide Web resources. The idea of using network technology in order to make the virtual communication laboratory is presented in the next section.

THE VIRTUAL LABORATORY USING MATLAB WEB SERVER

One of the important goals of the project is to make the laboratory resources available to students via Internet. The virtual laboratory should be accessible in an interactive fashion. It enables users to develop own algorithm with Matlab code or to run some finished applications under initial parameters chosen by them. The Matlab Web Server Toolbox meets these assumptions and enables administrator of the virtual laboratory to create MATLAB applications that use the capabilities of the World Wide Web to send data to MATLAB for computation and to display the results in a Web browser [3]. The entire communication process between Web application and Matlab works under control of a program called 'matlabserver'. It is located on the same computer where the Matlab application works, while the client program, called 'matweb' is installed on Web server. In the simplest configuration, a Web browser runs on student (end user) workstation, while MATLAB, the MATLAB Web Server, and the Web server daemon (for example Apache Web Server) run on another machine. The diagram showing how Matlab operates over the Web is illustrated in Figure 1.

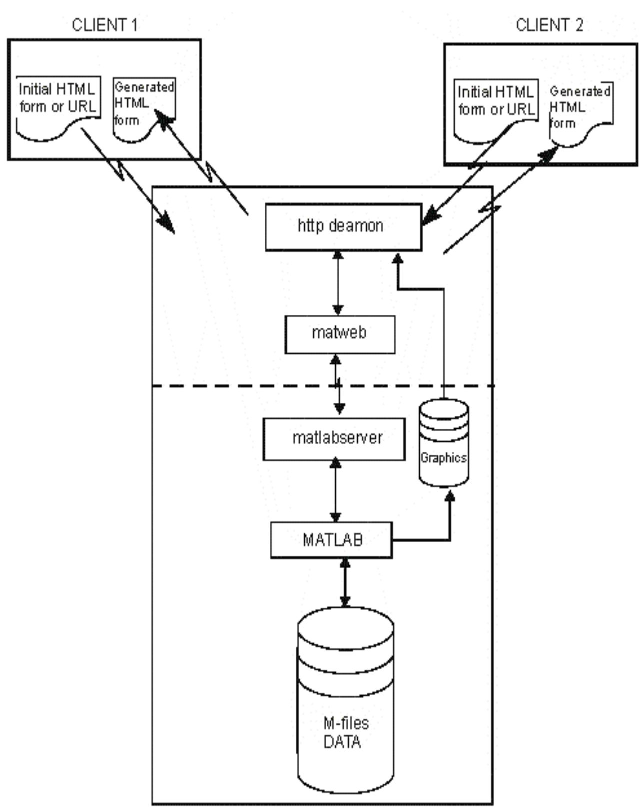

FIGURE. 1
MATLAB ON THE WEB (REPRINT FROM MATLAB WEB SERVER DOCUMENTATION)

How does it work from user's side? The end user, who wants to do some exercise, has to access the web page only. The opened HTML document is a form where user writes out simulations parameters or part of Matlab code. The filled form is sent into web server, where 'matweb' uses the Common Gateway Interface (CGI) to extract data from HTML documents and transfer it to the 'matlabserver'. The latter runs the m-files pointed out by HTML document with the extracted parameters. The results of calculation are sent to output HTML page and display at the user workstation.

Each experiment needs a separate set of HTML documents and m-files. The exercise's originator with the help of templates prepares all files necessary to do exercise. The Matlab Web Server Toolbox provides four templates, although the first three are only needed for full application:

- input_template.html – provides the code needed to create a Matlab Web Server input HTML document. There are several important lines of HTML code. These lines point out the program, that is responsible for extraction data from web page, and provide the name of the Matlab m-file to run.
- output_template.html – for creation an output document.
- mfile_template.m – provides the additional code needed to accept input from HTML input document and to return results to HTML output document. Variables transferred between web page and Matlab file are held in Matlab structures. We have one structure for every direction. When some plots are required in output HTML document, special function inserts them into JPEG standard files. Matlab can export maximum 256 KB of data to HTML output document for each m-file.
- tmfile_template.m – for debugging purposes.

Details, how each files can be constructed will be provided in the next section on the basis of exercise's example.

We should mention security issues in the end. They are done in simple way. The Matlab Web Server provides an optional configuration file, where administrator lists machines that could connect to the laboratory resources if some protection against unauthorized access were necessary.

EXAMPLE OF LABORATORY EXERCISE

A presentation of one of the laboratory exercises together with the virtual laboratory idea is shown in this section. 'Blind channel equalization' exercise will be used for example in order to better understand how the virtual communication laboratory will be made finally [4]. This example presents possibilities of teaching the latest technologies in convenient and user-friendly fashion.

The blind channel equalization is a popular approach to minimizing the linear channel distortions. These methods minimize the channel distortion without any training sequences from the transmitter. In blind equalization the receiver knows only the statistical properties of actual data sequence. Therefore the system efficiency is not decreased.

The best known blind method as Sato algorithm, Constant Modulus algorithm, Benveniste-Goursat algorithm and 'stop & go' algorithm are explored in this exercise. Complete execution of the exercise assumes that some skills were obtained in previous exercises. For example the LMS adaptation method, that is used in this exercise, is analysed in 'Adaptive filters and acoustic echo cancellation' in details. But the above suggestion doesn't have to get in the way of execution of the considered problem.

The entire exercise is divided into three independent parts as presented in Figure 2:

- Introduction
- Simulations examples
- Experiments

Every part has a form of a new page that can be called in a separate window.

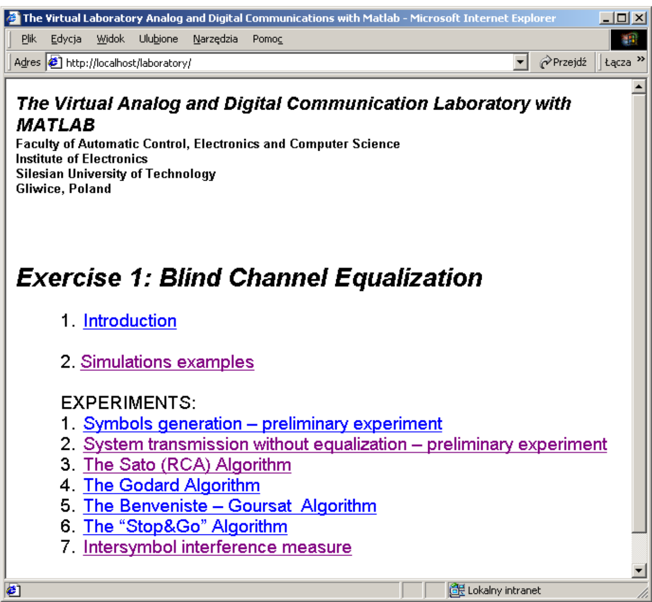

FIGURE. 2
THE TITLE WEB PAGE

Part One – Introduction

The comprehensive study into the blind channel equalization is the main objective of the 'Introduction' page. This ordinary Web page includes all information necessary to do the exercise. Each considered equalization technique is characterized on this page with help of an appropriate definition, equations and figures. If it is possible, some Internet links will be enclosed. Because this exercise relies on 'Adaptive filters', 'Channel equalization' and 'Quadrature Amplitude Modulation' exercises, some parts of information are only mentioned and user will be directed into these pages if he wants to seek detailed information. Next, there are described simulation's conditions (types of channels, equalizer's parameters etc.) and a set of hint on interpreting results that are obtained during simulations.

Part Two – Simulations Examples

This part of exercise gives deep insight into the considered problem by simulations. The web page is designed as a form in accordance with rules of Matlab Web Server application. The standard form has several spaces that enable users to choose or to write out his simulation parameters. An example of such form for the 'Blind Channel Equalization' exercise is illustrated in Figure 3.

This example depicts a convergence of the blind equalization algorithm depending on the adaptation method, type of modulation, the channel's characteristic and parameters of equalizer. The remaining ones as a number of symbols and signal to noise ratio specify all conditions of the simulations. Every possible combination of parameters is possible, although the satisfactory results are not obtained for all. When chosen parameters are proper, user clicks on the 'Perform' button and the data are sent through the 'matlabserver' to the given m-file.

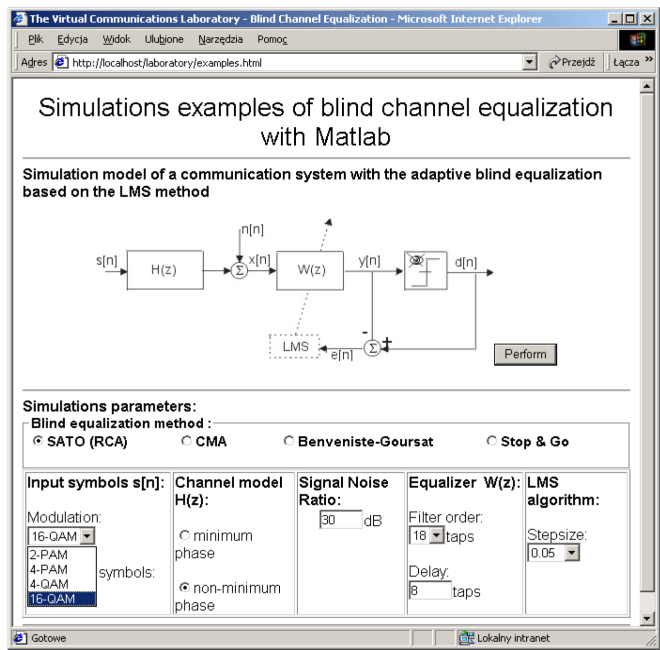

FIGURE. 3
THE SIMULATIONS EXAMPLE PAGE

Results of simulation are sent to another web pages. In a considered example, the output page includes only graphic results. The first three pictures, presented in Figure 4, depict the system parameters (signal constellation and channel characteristic) and the influence of the intersymbol interference on the received symbols. Of course, an effective reconstruction of them is impossible at this moment. Next two constellation diagrams depict effect of working of the blind algorithm, while the next picture shows the equalizer parameters achieved in the last step of adaptation. The last picture shows how fast the intersymbol interference level decreases for the blind algorithm (red line) in comparison with the training sequence algorithm (blue line).

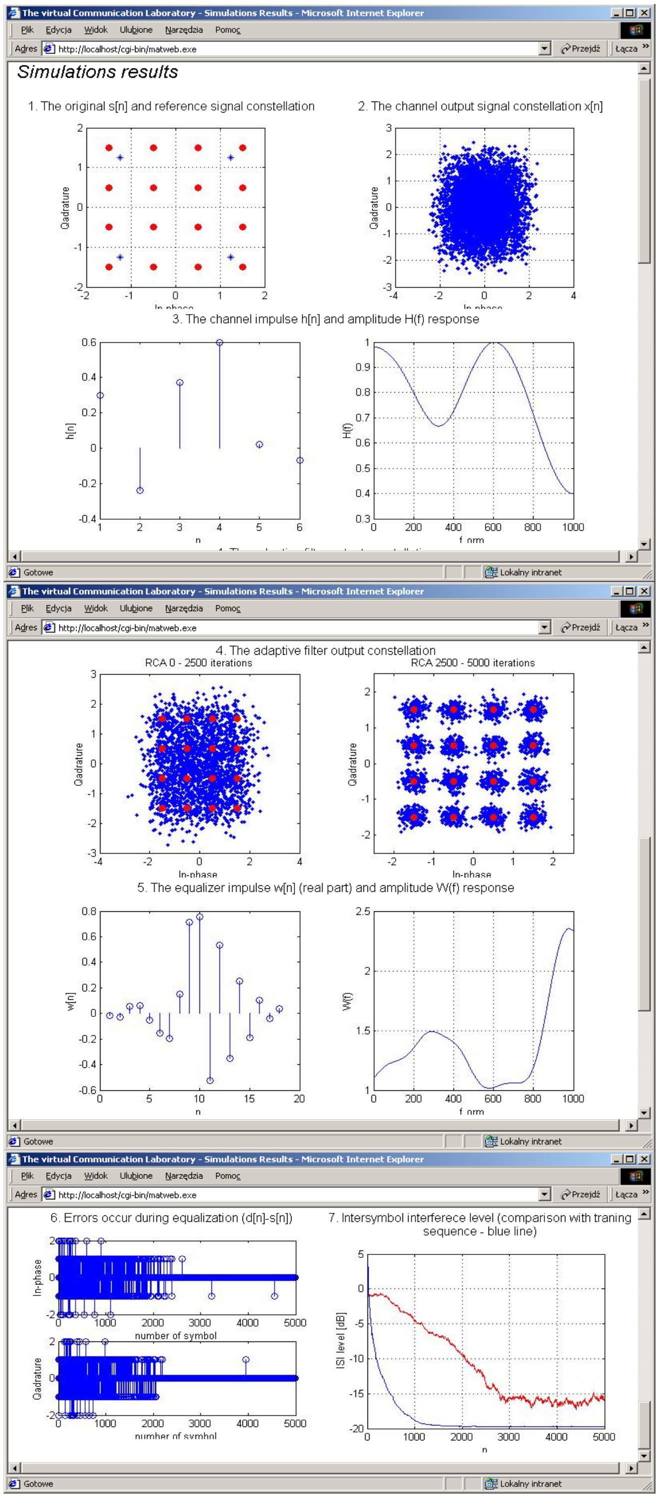

FIGURE. 4
THE RESULTS PAGE

The simulation page is the most simple interaction between user and Matlab. It frees them to create own Matlab applications. Eventually, such pages are suitable only for analysing some properties of the blind algorithms. If exploration of algorithm construction is required the next step should be done.

Part Three – Experiments

The previous part of the exercise doesn't give users any information about how communication algorithms work and can be made. It is very desirable know-how from engineers' point of view. Therefore, the 'Experiment' part places emphasis on code development. This simple assumption becomes more complicated when the virtual laboratory idea is taken into consideration.

The web page, designed for experiments purposes, has blank text spaces where user can write out his Matlab code. The code is treated by web pages as a string variable. Each text space for Matlab code is a different variable. These variables (in the form of Matlab structure) are the input parameters of m-file combined with a web page. Matlab uses special build-in function called EVAL in order to replace expression 'eval(name)' with the content of the string variable called 'name'.

The creation of the whole application in Matlab code using above approach is impractical, although it seems to be possible. Two main reasons proved this statement. Firstly, user should know not only standard Matlab build-in functions but also Matlab Web Server Toolbox ones. Secondly, even if he mastered it, he would have to know how the output web page is designed, because the output Matlab variables are strictly connected with HTML documents.

Therefore in laboratory experiments users can develop only the parts of an application code. They are specially selected fragments of the entire application, which are crucial to understand and gain skills at considered area. The example of web page for developing some blind equalization algorithm is presented in Figure 5. In comparison with the input page of simulations examples, this one has an additional blank text spaces where can be written out proper code. Each blank space is precisely described. It contains short description how the function should work and all input and output parameters. The correct work of the made function can be ensured, if user uses the same name of input and output variables as it is described in the web page. Otherwise any error doesn't allow running Matlab application.

In presented example, user has to develop two functions (signal generator and adaptive RCA-LMS algorithm), and characterize impulse response of the communication channel using Matlab code. Additionally blank text spaces in 'Plots' part enable users to describe which variable will be depicted as a figures in the output web page.

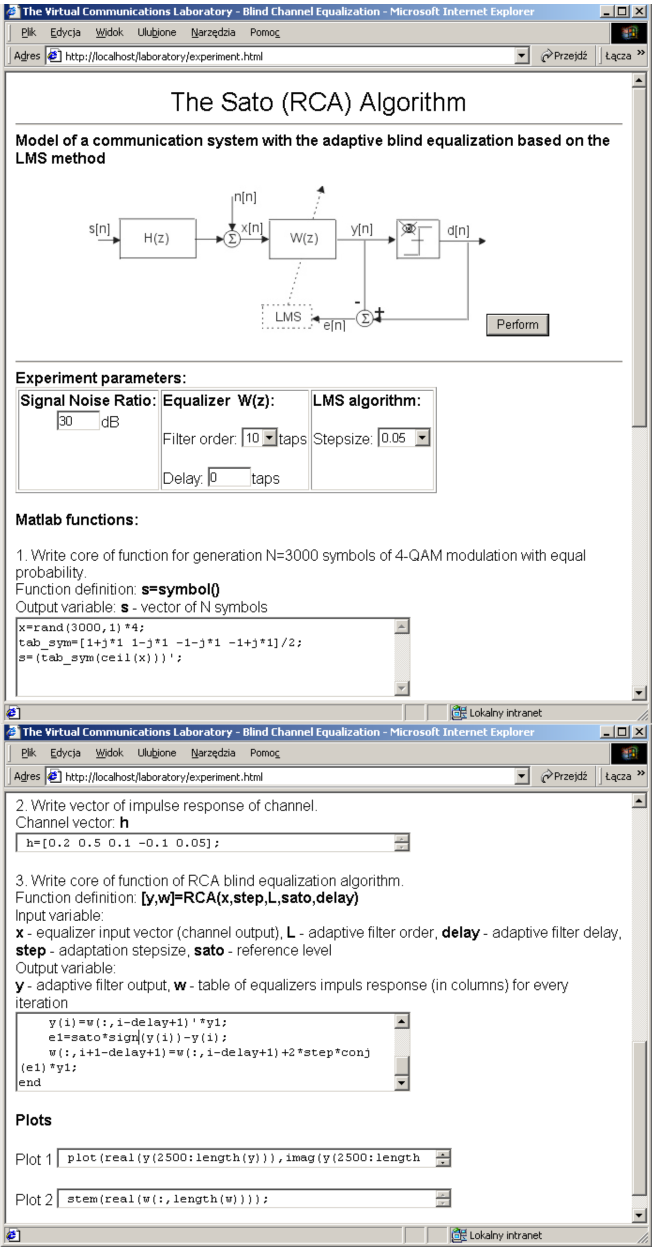

FIGURE. 5
THE EXPERIMENT PAGE

CONCLUSION

The virtual communication laboratory using Matlab over the Internet was presented in this article. The simplification of education process and presentation of the wide scope of communication technologies using on-line access to laboratory resources are the main objective of this project, that is currently under construction in Institute of Electronics, Silesian University of Technology in Gliwice, Poland. The entire number of exercise will come to about 35. It means that number of web pages with examples and experiments is estimated above 100. Current efforts are aimed to prepare experiments in Matlab environment, next to create web pages for remote access to laboratory resources.

REFERENCES

[1] Izydorczyk J., Płonka G., Tyma G., "Teoria sygnałów", *Wydawnictwo Helion,* Gliwice, 1999.

[2] Sklar B., "Digital Communications", *Prentice Hall PTR,* New Jersey, 2001.

[3] Matlab Web Server – *Matlab documentation*, Version 1.

[4] Dziwoki G., "The Blind Channel Equalization", *laboratory instruction of "The Fundamentals of Analog and Digital Communications"(be in preparation).*