15th Triennial World Congress, Barcelona, Spain

A DISTANCE LEARNING COURSE IN MODELLING AND SIMULATION

Jukka Lieslehto, Mikko Hakala, Tero Karppinen

Institute of Automation and Control
Tampere University of Technology
P.O.Box 692, FIN-33101 Tampere, Finland

Abstract: The paper describes web-based distance learning material for a course in modelling and simulation. The course covers topics like systems and models, physical modelling, system identification, and simulation. An essential part of the material are interactive demos that are implemented with Matlab, Simulink and Matlab Web Server. *Copyright © 2002 IFAC*

Keywords: Education, identification, modelling, simulation, teaching.

1. INTRODUCTION

The rapid development of computer and communication technology, especially Internet and World Wide Web, has made a new kind of education possible. Distance learning is supplementing, or even replacing, traditional teaching methods in all educational levels, from primary schools to universities.

In the field of control education both web based remote laboratories (Miele et al., 2001; Röhrig and Jochheim, 2001) and control courses (Bark et al., 1999; Copinga et al., 2001) have been introduced. The subject of this paper falls into the latter category.

The goal of the course Modelling and Simulation is to give readiness to perform tasks involving modelling and simulation of dynamic systems. The course is based on the textbook Modeling and Simulation of Dynamic Systems by Ljung and Glad (1994). It is a one-semester course aimed at third year students, and it consists of 28 hours of lectures and 28 hours of computer exercises. The lectures and exercises are once a week and both take two hours at a time. In the computer exercises students learn how to build simulation models for simple dynamic systems using Matlab (Mathworks, 2000a) and Simulink (Mathworks, 2000b). The purpose of the developed distance learning material is to supplement, not to replace, traditional teaching.

The course is one of the pilot courses in a national distance education project. Its selection as one of the pilot courses was motivated by several reasons. Number of students taking the course is usually rather large, typically over fifty. This can be a problem because teaching resources are limited. The course is also in the curriculum of the conversion education program offered by the university. The participants of the program are B.Sc. engineers who aim to upgrade their degree to M.Sc. They are usually working in industry, and thus they often miss lectures and exercises. The third reason was the subject of the course. Because it is a computer-oriented course, it is also a natural test case for new computer-based teaching methods.

This paper is organized as follows. Section 2 describes the software architecture of the system. Section 3 deals with the structure of the course from a content point of view. Feedback from students is summarized in Section 4, and conclusions finish the paper.

2. SYSTEM ARCHITECTURE

A straightforward way to create a distance learning course is to build a web site consisting of dozens of web pages written in Hypertext Markup Language (HTML). However, this is not an adequate solution for a course like Modelling and Simulation due to two reasons: course material is mathematically oriented and interactive demos are an essential part of the course.

2.1 Mathematics

Only very simple mathematical equations can be written using HTML. More complex equations are often included as images. However, this is not a flexible solution if the number of equations is large.

Another solution would have been to use Mathematical Markup Language (MathML) which is intended to facilitate the use of mathematical and scientific content on the Web. Unfortunately, MathML is still not a mature technology. This means, for example, that browsers can not render MathML without special plug-ins.

Our solution was to use Adobe Portable Document Format (PDF) when a page includes complex mathematical equations. PDF files can be shared, viewed, navigated, and printed by anyone with free Adobe Acrobat Reader software. PDF is probably the most widely used format for mathematically oriented content on the Web.

2.2 Dynamic content

The web pages of the course are partly in HTML, and partly in PDF format. The remaining problem is that these are both formats for static web content. Dynamic content in the form of interactive demos is important for the course. One can add dynamic content to web pages using languages like Perl, Java, and JavaScript. However, this can take a lot of effort because those are more or less general purpose languages that do not specifically support modelling and simulation applications.

Recently, solutions have been proposed that make it possible to run Matlab and Simulink applications from web pages (Hassan et al., 1999; Mathworks, 2000c). Matlab and Simulink are widely used and well suited for modelling and simulation.

The interactive demos of the Modelling and Simulation course are implemented using Matlab, Simulink, and Matlab Web Server software. System architecture is depicted Fig. 1. Matlab and Matlab Web Server software run on a server. Several copies of Matlab can run concurrently. Clients communicate with the server through their web browsers. Clients do not need any special software besides a web browser.

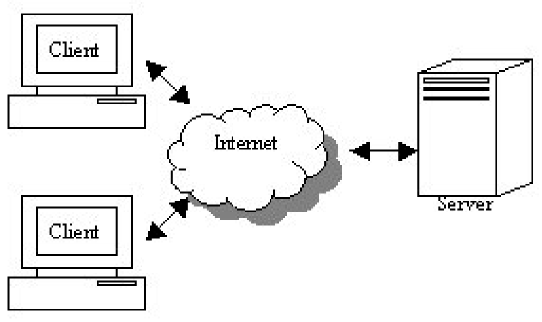

Fig. 1. System architecture.

Interactive demos typically consists of two HTML documents and one Matlab M-file. The first HTML document is for collecting data from a user. The data is collected using HTML forms. Then the data is send to the server, where it is extracted from HTML documents and transferred to a Matlab program (M-file) which performs necessary numerical computations. An HTML document containing the results is generated by the Matlab Web Server. This document can contain both text and images. The document is send to the client and displayed in its web browser. Fig. 2 shows how data is moved between the clients and the server.

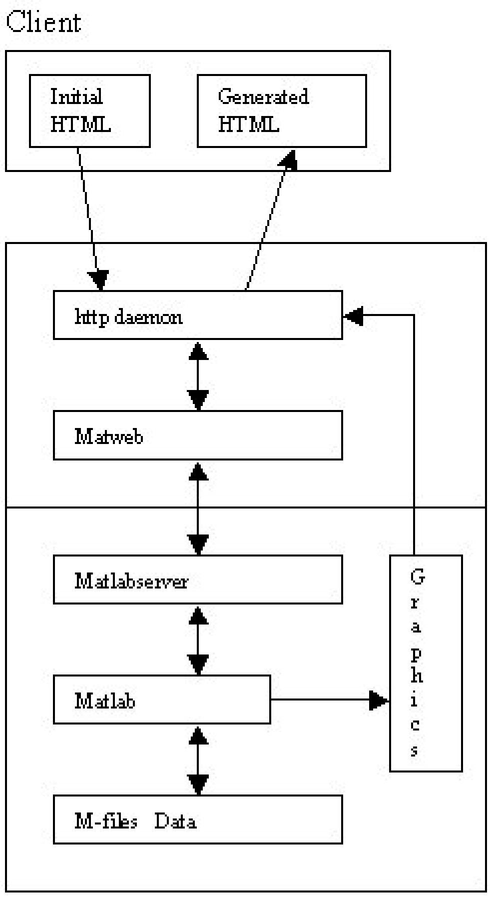

Fig. 2. Communication between client and server.

3. STRUCTURE OF THE COURSE

Like the book, the course and the web material are divided into four parts:

- Models. The first part deals with typical ways of describing dynamic systems.
- Physical modeling. The second part discusses how to construct a mathematical model from knowledge of the basic mechanisms of a system.
- Identification. The third part deals with methods to arrive at mathematical models from input/output data measured from a system.
- Simulation. The fourth part describes how models are simulated.

3.1 Models

The navigation map for the first part of the material is shown in Fig. 3. Each block represents a model type, and each arrow a transformation between two model types. The upper part of the map describes time continuous models, and the lower part time discrete models. Color codes are used to separate non-linear and linear time domain models and frequency domain models. A student can find information on a model type or transformation by clicking the corresponding block or arrow. Because the material is mathematically oriented including several rather complex equations, the pages giving detailed descriptions of model types and transformations are in PDF format.

Fig. 3. Navigation map of models.

3.2 Physical modelling

In this part students can go through the different phases of the physical modelling process: structuring the problem, setting up the basic equations and forming the state-space model. The theoretical content of the web pages is very limited because a more detailed presentation is given in the book and lectures. The emphasis is on examples which go through the phases of physical modelling.

Students can simulate the models created by physical modelling with different parameter values. The parameter values are send to the server where Matlab and Simulink are running. There Matlab runs a simulation and sends the results back to the client where they are displayed in a new browser window. Simulation results are presented in graphical form, and from these figures a student can easily see how the changes in parameters change the simulation results.

3.3 Identification

A navigation map of the identification cycle, shown in Fig. 4, is the initial page of the identification part. From the map students can easily see the main phases of the cycle and how they depend on each other. The elements of the map work as links to pages which have more detailed information. The shape of the element tells whether the user (oval) or computer (rectangle) has the main role in that phase.

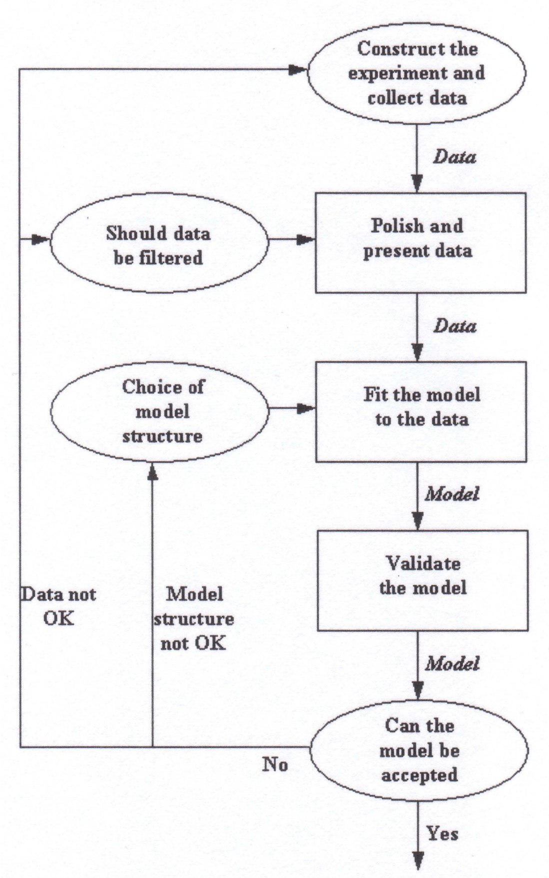

Fig. 4. Navigation map of the identification cycle.

The pages which have more detailed information of each subject contain some theory, examples, and interactive demos. The theory covers only the main points of the subject because these questions are covered more thoroughly in the book and lectures. Also the examples focus on the practical side of identification.

In the interactive demos the user can estimate the impulse response of a system using either transient or correlation analysis. Frequency response estimates can be computed using Fourier or spectral analysis. Also the most typical parametric models, ARX, ARMAX, OE, and BJ, can be estimated. Interactive demos are also available for a selection of model validation methods: comparison of measured and predicted outputs, residual analysis, comparison of parametric and non-parametric models, pole-zero -plots and estimates of parameter uncertainty.

All computations are performed on a server computer which generates a web page containing the results and sends it to the client computer. Matlab's System Identification Toolbox (Ljung, 2000) is used for computations.

3.4 Simulation

The fourth part deals with simulation. The main focus is on numerical integration algorithms. The material covers topics like fixed time step vs. variable time step, one-step vs. multistep methods, explicit vs. implicit algorithms, low-order vs. high-order methods, and stiff vs. non-stiff differential equations. In interactive demos students can simulate a given system using different numerical integration algorithms, like Euler, Runge-Kutta, and Adams-Bashforth. It is possible to use either fixed or variable time steps.

4. STUDENT FEEDBACK

During the development of the material feedback was collected from students who had taken the course earlier. That was a good way to find out the main problems and see how different students understand various things in the material. The feedback from testers was very good and they considered the web site to be easy to use. They liked the idea that users can independently choose what they want to do and in which order. Ease of navigation is also considered very important, and this emphasizes the logical structure of the site.

The material was in actual use for the first time during the autumn semester of 2001. Feedback from the students was mainly positive, although some technical problems were encountered. Matlab Web Server occasionally stopped working, which meant that at these times the interactive demos were out of use. This was, naturally, a cause of frustration for some students, although the rest of the material was still available. A summary from the student feedback is that it is a good idea to make some material available on the web, but it is, at least at the moment, impossible to totally replace traditional teaching, like lectures and exercise, with distance learning material.

5. CONCLUSIONS

In the field of education, distance learning can be used in two different ways: it can replace traditional teaching or it can support it. If it replaces the lectures and face-to-face exercises, material has to be large enough and has good composition. Even then, it has to be possible for a student to contact a teacher, for example by e-mail, and ask questions and advice.

Partly due to limited resources, time and money, a supportive role was chosen for distance learning material in this project. If the education on the Web just supports traditional teaching, there is no need to give all information on the Web. The teachers and students can still meet face to face at the lectures and exercises but the Web is a good extra tool in education.

The material of the Modelling and Simulation course is mathematically oriented. This meant that the traditional way of building a web site as a collection of web pages written in HTML had to be extended. Static content containing several complex mathematical equations can be easily created and distributed on the Web using the PDF format. Dynamic content, like interactive demos, were created using Matlab, Simulink, and Matlab Web Server software. This was considered a more flexible solution than the use of "web-oriented" languages, like Java, because Matlab and Simulink are high-level languages that support modelling and simulation extremely well.

Because of the supportive role of the material, examples and interactive demos are emphasized in it. It is our opinion that textbooks and traditional lectures are more efficient ways to present theoretical content than the Web.

REFERENCES

Bark, A., B. Egardt and B. Kristiansson (1999). A Web-based Course in Automatic Control for Engineers in Industry. *Proc. of the 14th IFAC World Congress*, Vol. M, 235 – 239.

Copinga, G.J.C., B. De Schutter and J.M.A. Scherpen (2001). A Web-based Study Support Environment for Systems and Control Courses. *Proc. of the American Control Conference*, 1155-1160.

Hassan, C., R Tuschák, I. Vajk, R. Bars, J. Hetthëssy, F. Kovács & G. Szitnyai (1999). A New Web/Matlab Based System in Control Education, *Proc. of the 14th IFAC World Congress*, Vol. M, 241 - 246.

Ljung, L. (2000). *System Identification Toolbox for Use with Matlab (Ver. 5).* Mathworks.

Ljung, L. & T. Glad (1994). *Modeling of Dynamic Systems*. Prentice - Hall.

Mathworks (2000a): *Using Matlab (Ver. 6).* Mathworks.

Mathworks (2000b): *Using Simulink (Ver. 4).* Mathworks.

Mathworks (2000c): *Matlab Web Server*. Mathworks.

Miele, D.A., B. Potsaid & J. T. Wen (2001). An Internet-based Remote Laboratory for Control Education, *Proc. of the American Control Conference*, 1151-1152.

Röhrig, C. & A. Jochheim (2001). Group-based Learning Using a Remote Laboratory, *Proc. of the American Control Conference,* 1153-1154