Interactive applications in teaching with the MATLAB Web Server

Article · January 2001

CITATIONS
9

READS
218

2 authors:

Andreas Pester
The British University in Egypt
63 PUBLICATIONS **329** CITATIONS
SEE PROFILE

Ramiz Ismailov
Alpen-Adria-Universität Klagenfurt
1 PUBLICATION **9** CITATIONS
SEE PROFILE

Interactive Applications in Teaching with the MATLAB Web Server

Andreas Pester, Ramiz Ismailov

Carinthia Tech Institute, School of Electronics

Key words: *Tele-learning, MATLAB, Matlabserver, Web-based simulation*

Abstract:

The MATLAB Web Server is a special toolbox to access MATLAB-Simulations from the web. We installed this server in the environment of a HyperWave Information Server and got good results using this tool for teaching electrical engineering and mathematical modeling. The main advantages are: on-line simulations, working only through a browser-interface with MATLAB, simulation with the user's own data via the Internet. The main disadvantages are: slow Internet access time for movie-simulations, not possibility to use GUI's and no flexibility for the models. The only user-chosen inputs are the data. We described some possibilities to solve these problems.

1 Aim and structure of the MATLAB Web Server

The MATLAB Web Server is a new toolbox for the well-known software-product **MATLAB®** of the MathWorks, Inc. (http://www.mathworks.com). This tool is a web-front-end for MATLAB and it is possible to design MATLAB-simulations running on a special server started from a client. The output of the simulations is shown on a user-designed html-document of a web browser.

Connection between client application and MATLAB is shown in Figure 1. Client application, usually a web browser, loads an "Initial" Hypertext Markup Language (HTML) [1] document from Hypertext Transfer Protocol [2] Daemon (HTTPD). After editing the parameters is completed, the client sends data to the HTTPD, which loads matweb through Common Gateway Interface (CGI) [3]. Matweb connects to the matlabserver by means internal protocol. Matlabserver loads the requested M-file into a separated copy of MATLAB. After executing has been completed the matlabserver gives an output to the matweb, usually as text/html based on a template file. The matweb returns it to the client through HTTPD. During execution of an M-file, image-files, extra html-files or any other data files can be created and linked to the main output html. In this case the browser will download them from HTTPD. MATLAB also supports Java classes [4]. By using Java classes on the server as well as on the client side it is possible to extend the default connection schema.

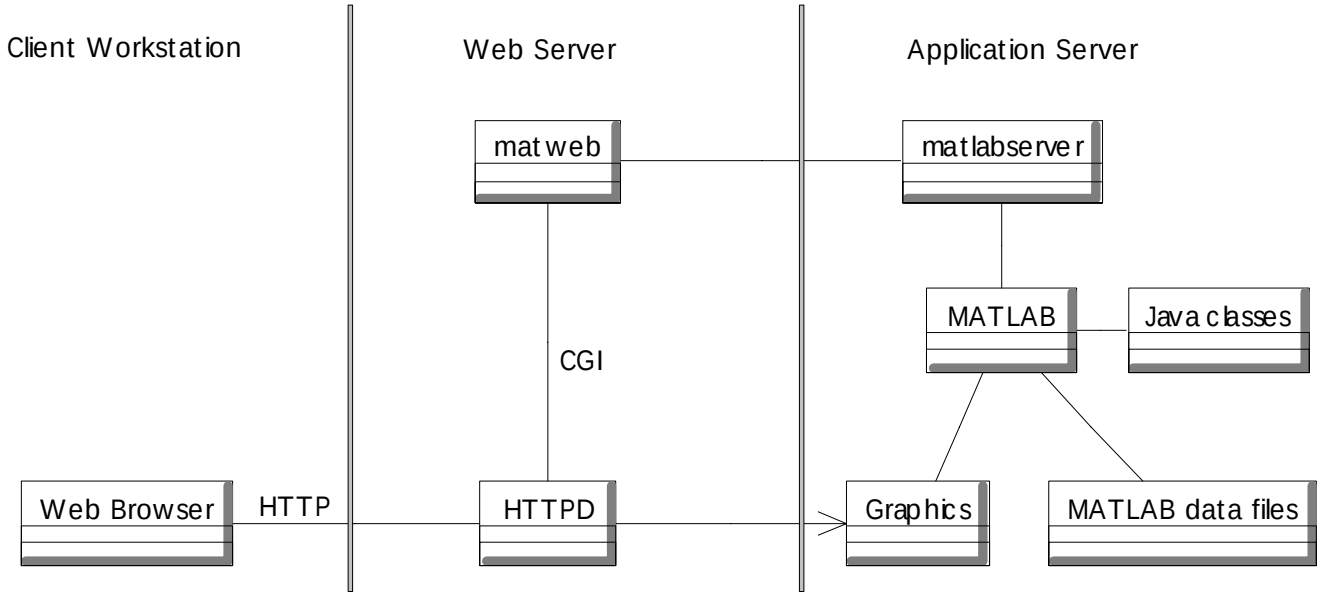

Figure 1. *Logical model of MATLAB Web Server*

To use the MATLAB Web Server MATLAB, the toolboxes needed for the MATLAB-scripts, the toolbox MATLAB Web Server and a Web-Server must be installed. Any HTTPD that supports CGI can be used.

MATLAB R12 is available for platforms Win32 (Windows NT 4.0, Windows 2000)[5] and UNIX (Solaris [6], Linux [7], etc.).

As shown in Figure 1, performance of MATLAB Web Server is dependent upon performance of HTTPD. Additional to supporting CGI requirements, the HTTPD must support the so-called “Virtual directories”. This is important because the MATLAB Scripts (M-files) have to be located in subdirectories of the MATLAB root directory. On the Win32 platform the HTTPD can be the Internet Information Server 4.0 (or later) for Windows NT 4.0 or Internet Information Services 5.0 [8] for Windows 2000. On the UNIX platform the Apache HTTP Server 1.3 [9] can be chosen as the HTTPD. On the Linux platform -- beginning with kernel-2.4.0 [10] -- response time while loading large media files through HTTP can be reduced if module KHTTPD-HTTPD built in OS kernel is used. KHTTPD is designed to deliver files quickly through HTTP. It can also provide translucent-call to an external HTTPD, for example, to an Apache HTTP Server if it executes CGI or another server extension.

To design a MATLAB-Simulation that runs on a MATLAB Web Server, experience in html programming and in MATLAB-Programming is needed. Input- and output-documents in html and one or more MATLAB-functions (running on the server) have to be designed. Additionally the application needs to be registered in a configuration file (by hand). Automatic registration is not provided.

2 Facilities and advantages of using the MATLAB Web Server in teaching

With the MATLAB Web Server real time simulations on-line can be done. These are needed in many technical courses and in particular virtual laboratory environments. In addition they are useful in labs for mathematic courses. The input in the simulation interface can be

numerical or alphanumerical and the output can be numerical, alphanumerical or graphical. As usual in MATLAB, the numerical input values can be real or complex (See: http://ml.cti.ac.at:5050/webmagic1.html or http://hwis01.cti.ac.at:5050/cplxroots1.htm). If an alphanumerical output is necessary, familiarity with the MATLAB-Programming in strings is needed. For an example of the MATLAB Web Server being used as a calculator for Laplace transforms, see: http://ml.cti.ac.at:5050/weblaplace1.html. However, symbolic calculations are not the most effective method of using the MATLAB Web Server.

For the most effective way to use the MATLAB Web Server in engineering or electronic courses is 2D graphics, see http://ml.cti.ac.at:5050/VanDerPol_11.html and for 3D graphics, see http://hwis01.cti.ac.at:5050/Startseite.htm. Some examples from these web pages are: 2D plot of a phase-portrait of the Van der Pol differential equation (Fig.2) and a 3D plot of wire resistance (Fig. 3).

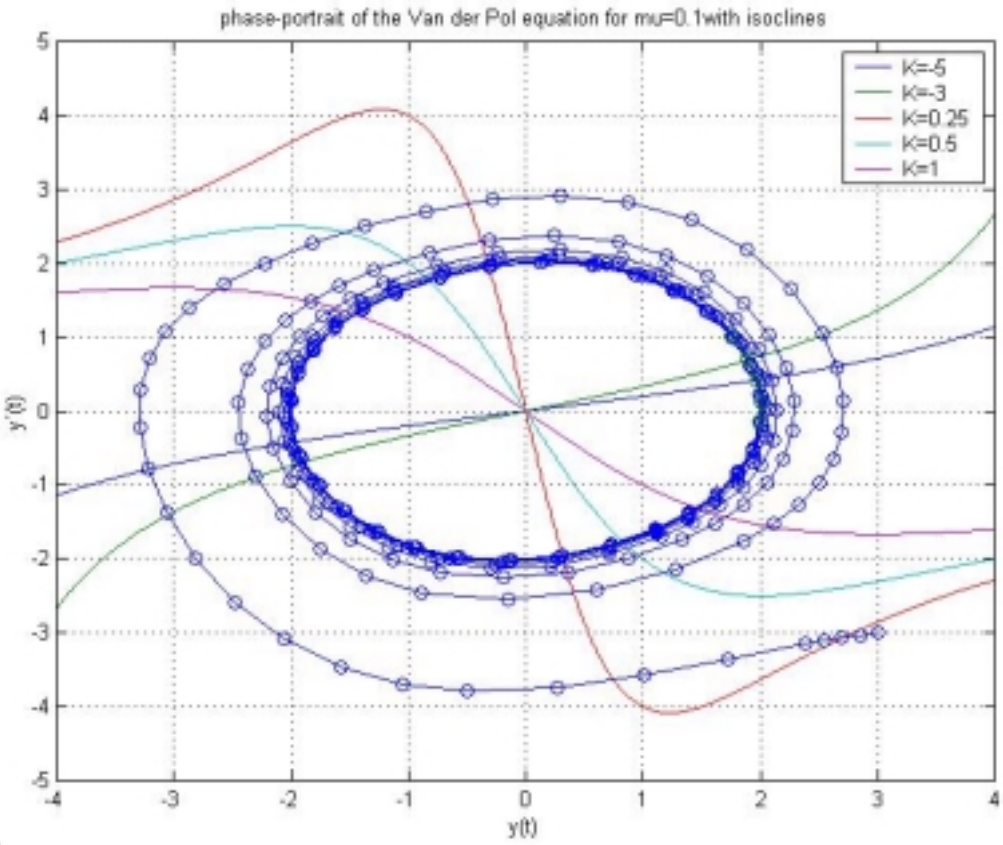

Figure 2

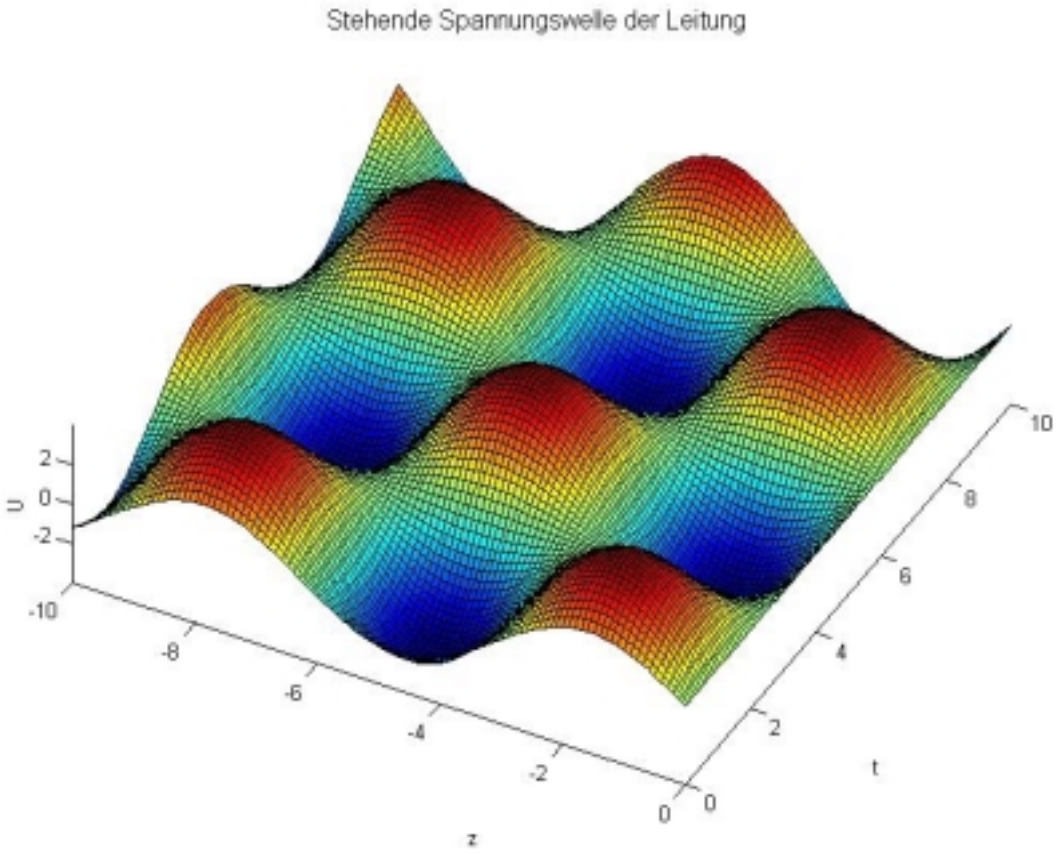

Figure 3

The most important advantage of the MATLAB Web Server is that users can use MATLAB-Applications without installation of MATLAB-Software on their client computer. Any HTTPD interface is enough to run MATLAB Web server applications, for example, a web-browser. The users also don't need to have special knowledge of MATLAB and MATLAB-Programming to run the chosen simulation. Some companies take advantage of this facility by promoting their software-products with web-based simulations and models running via a MATLAB Web Server (http://www.femlab.com). From a teaching and learning point of view this advantage is only interesting in that it saves time and increases student motivation. Often there is insufficient time for a trainer to re-write a MATLAB script for every visualization or simulation, and in addition the trainer is independent from the software environment installed on a client computer.

From a didactical point of view, it is important that users (students) can try all the models with their own parameter-values. Re-adjustments are needed only in cases where a valuable didactic visualization depends on the input values of the parameters. For students in the 3rd semester it was very interesting to model and program problems from mathematic and electrical engineering with the MATLAB Web Server (http://hwis01.cti.ac.at:5050/RCL_gekoppelt.htm). The numerical model to calculate this model is based on Laplace-transform and the numerical solution of an ODE system.

From a technical perspective, the main point is that the use of MATLAB Web Server applications is independent from the OS used on the server side. There is no need for

application maintenance by the clients. This is done only on the server and also the program upgrade. There is no need for an additional plug-in.

Since the MATLAB release R12, MATLAB Web Server can be extended with Java classes. This feature allows using Java-based libraries and technologies, such as Java Database Connectivity (JDBC) [11] or Java Servlet Technology [12].

MATLAB Application Server is accessible via standard HTTP protocol. This feature allows connection to MATLAB not only from Web browsers, but also from applications that implement this protocol. Thus it is possible as Java applets [13] and AcitveX controls [14] embedded in html documents as stand-alone applications.

3 Disadvantages of using a MATLAB Web Server

A serious disadvantage of the MATLAB Web Server is that, although it is possible to program 3D animations in MPEG- or AVI-format, the response time of applications is too long (1 min. or more). This is impractical considering current Internet conditions. This means there isn't any way to show processes in their extensions in time or space, only the end-result of a process. This narrows the value of the simulations.

Also MATLAB-GUI's are not useable with the MATLAB Web Server. In this case you can avoid the problem by designing the html input document in an adequate manner, but the use of sliders for continuous extensions of some input-values (common to MATLAB-GUI) is not possible.

For the alphanumerical input there are problems with the use of signs like '+' or '%' or '&', because they are control values for the HTTP-protocol. A solution is to use various scripts that replace these signs with others not used in the HTTP-Protocol, and to redo these changes in the MATLAB-script on the server.

One of the disadvantages of MATLAB Web Server is that it requires considerable hardware. Because MATLAB is an interpreter, it needs a high performance CPU and also a lot of RAM. Hardware requirements of the CPU are dependent upon the platform on which MATLAB is running. For an Intel platform, an Intel Pentium II or higher is strongly recommended. RAM consumption also depends on the platform. On the Windows NT 4.0 Server SP6.0a, matlabserver consumes 13MB of RAM. Additionally, each MATLAB running under without memory for data consumes 25MB of RAM and 37MB of VM. On RedHat Linux 7.0 kernel-2.4.1, each matlabserver consumes 1.7MB of RAM. Each MATLAB running under matlabserver consumes 22MB of RAM.

Before executing, MATLAB compiles M-file into pseudo-code. This operation takes place only once when the script is called the first time, then the compiled code is cashed and used on the next calls. It is also possible to force MATLAB to compile M-files and store pseudo-code into P-files to reduce response delay of the first call. It is also possible to translate MATLAB script into C/C++ code and then compile it in native executable code or in a dynamic loading library.

From a didactical point of view a disadvantage is that there are only useable models with fixed structure. The user can only change the input values, but not the whole model. In some cases the change of the model structure would be very helpful (Think about symbolic and

numerical solutions of ODE). One way to avoid this problem is to use MATLAB on an application-server or on a terminal-server via TCP/IP protocol. There will be no licensing problems because the user-groups for such servers are limited.

References:

[1] W3C HTML Specification – http://www.w3c.org/MarkUp/

[2] RFC 2616: Hypertext Transfer Protocol – HTTP/1.1

[3] W3C CGI Specification – http://www.w3c.org/CGI/

[4] Java language specification – http://java.sun.com/docs/books/jls/

[5] Microsoft Windows – http://www.microsoft.com/windows/

[6] Sun Solaris – http://www.sun.com/solaris/

[7] Linux official homepage – http://www.linux.org/

[8] Microsoft Internet Information Services – http://www.microsoft.com/iis/

[9] Apache HTTP Server Project – http://httpd.apache.org/

[10] The Linux Kernel archives – http://www.kernel.org/

[11] JDBC – http://java.sun.com/jdbc/

[12] Java Servlet Technology – http://java.sun.com/servlet/

[13] Java Applet Technology – http://java.sun.com/applets/

[14] ActiveX Controls – http://www.microsoft.com/com/tech/activex.asp/

Authors:

Prof. Dr.phil. habil. Andreas Pester (A.Pester@cti.ac.at)
Carinthia Tech Institute, School of Electronics
Richard-Wagner-Strasse 19
A-9500 Villach

Ramiz Ismailov (ramiz@ukr.net)
National Technical University (KhPI)
Frunze street 21
Kharkiv 2
61002
Ukraine