Remote Web operation of an inverted pendulum

Manuel Rodrigues Quintas\textsuperscript{1,2}, António Mendes Lopes\textsuperscript{1,2}, Carlos Moreira da Silva\textsuperscript{1,2}, Paulo Abreu\textsuperscript{1,2} and Diego Sá\textsuperscript{1}

\textsuperscript{1} Universidade do Porto/Faculdade de Engenharia, Porto, Portugal
\textsuperscript{2} IDMEC – Pólo FEUP, Porto, Portugal

Abstract — This article presents the developments made in the implementation of an inverted pendulum test rig with the possibility to be remotely operated throughout the Web. This laboratory experimental system was chosen due to its intrinsic rich nature in terms of non-linear control theory. It provides an experimental platform to study, develop and test different control strategies and control laws. The article starts with the description of the electromechanical system, including both the mechanical components and the sensing systems. The global architecture of the control system is then presented. The pendulum is controlled by a Personal Computer, using the Real-time Windows software available within the MATLAB/Simulink application. It is then described the graphical interface developed for local operation of the inverted pendulum. The user interacts with the system through this interface that enables the command and monitoring of the system. The interface software was developed using the C# language present in .Net Framework and communicates with the MATLAB/Simulink application through MATLAB Application Type Library. To remotely operate and monitor the system, a Web application was developed, using a server/client architecture. On the client side an application based on JAVA (applet) technology implements the graphical interface with the remote user. On the server side, it was developed a dedicated application using the same hardware/software platform used for the local graphical interface. Finally, the conclusions are presented.

Index Terms — MATLAB/Simulink, Real-Time operating system, Remote laboratories, Web application.

I. INTRODUCTION

An inverted pendulum is a classical mechanical system that has been widely used in the demonstration of theoretical and practical aspects of control theory. The intrinsic non-linearity and unstable characteristics of the inverted pendulum are used to study the problem of stabilization of such systems and to validate different control strategies. Having this in mind, it was developed at the UISPA (“Unidade de Integração de Sistemas e Processos Automatizados”), IDMEC – Pólo FEUP, School of Engineering University of Porto (FEUP), Portugal, an inverted pendulum assembled on a moving carriage powered by an electric motor, having a control architecture totally opened (Fig. 1). The user can interact with the system, locally or remotely (through the internet), using graphical interfaces developed with identical functionalities. This capability of remote interaction with the system increases significantly the number of potential users, the security conditions and the availability of the system, while enabling an economy of resources.

The implementation and the availability of remote experiments and remote laboratories had a large increase during the last few years due to the rapid expansion of Internet and associated technologies. These facts opened a window of opportunity, particularly in teaching institutions, to enlarge their target public, share resources, saving on equipments and human resources, as well as to create an additional motivation to their students.

In designing the control system of any physical system, one of the most important aspects to consider lies on the selection of the real-time environment necessary to run the control application. If the hardware is based on personal computers, the main alternatives are the use of real-time operating systems (e.g. QNX, VxWorks, VRTX), real-time extensions for non real-time operating systems such as Windows and Linux (e.g. RTX, Intime, RTLinux, RTAI) or using applications integrating real-time modules with simulation and development environments (e.g. MATLAB/Simulink, LabVIEW). These later programming systems present clear advantages when the main objective is the development of an open control system, where the control algorithms and parameters must be easily modified, simulated, implemented and tested. Furthermore, the development environment that this type of application offers is user-friendly. Typically, they have real-time device drivers to interact with a wide variety of input/output boards and enable the development and/or integration of graphical interfaces, build on other standard environments (e.g. Visual Basic, C/C++, C#).

Figure 1. Inverted pendulum test rig
When the request to interact remotely with the physical system is present, it is also necessary to choose the appropriated technology. There are several solutions based on LabVIEW [1], and on LabVIEW/MATLAB/Simulink where the system control is based on MATLAB/Simulink while the graphical interface and Web integration are supported by LabVIEW [2]. The use of just MATLAB/Simulink has the advantage to incorporate in the same environment the simulation, the real-time control and the Web integration ([3, 4, 5, 6]).

The present article describes the electromechanical and control systems, as well as the graphical interface developed to operate locally or remotely an inverted pendulum. The pendulum is controlled by a Personal Computer, under a real-time environment provided by the Real-Time Windows Target available within the MATLAB/Simulink software package from MathWorks. This application (with additional toolboxes) is also used as a teaching tool and as the development platform for control algorithms. The user interacts with the system through a graphical interface specially developed that enables the command and monitoring of the system. This interface was developed in the .Net Framework, using the C# language, and communicates with the MATLAB/Simulink application thorough MATLAB Application Type Library. For remote operation and monitoring of the system, a Web application was developed, using a server/client architecture. On the client side it was developed an application based on JAVA (applet) technology that implements the graphical interface with the remote user. On the server side, it was developed a dedicated application using the same hardware/software platform used for the local graphical interface.

This paper is organized as follows. Section II describes the electromechanical system, namely the mechanical transmission, the driving and sensing systems. In section III it is presented the global architecture of the control system. Section IV describes in detail the control application, including the state machine implemented to control and interact with the controlled system. In section V it is presented the application that enables the remote operation of the system. Finally, in section VI the conclusions are presented.

II. ELECTROMECHANICAL SYSTEM

The experimental setup is composed of a pendulum that freely rotates around a horizontal axis mounted on a guided translating carriage. A brushless servo motor drives the horizontal translation motion of the carriage, through a gearbox, pulley and timing belt transmission chain. The pendulum consists of a 500 mm oscillating rod along which a 0.5 kg mass can be positioned. The total pendulum mass is 1 kg. A Heidenhain E RN 1020 incremental encoder mounted on the pendulum axis measures the angular position of the rod. The angular velocity is obtained by differencing the position signal.

The pendulum suspension point is mounted on a platform that moves with very low friction and allows the adjustment of backlash up to 10 mm relative to the carriage. This is a particular and important feature of the mechanical system. This adjusted backlash introduces perturbations, giving room to the development of control strategies that accommodate these uncertainties. A Solartron M922943A425-1 LVDT transducer is used for the measurement of this backlash.

The carriage moves along an 800 mm effective working displacement delimited by two inductive sensors. Measurement of the horizontal position of the carriage is obtained by means of the servomotor incremental encoder. An inductive sensor located at the middle of the working displacement is used as zero reference.

A Parker Hannifin Corporation ML2340 brushless servo motor drives the carriage up to a 1.25 m/s velocity obtained with a mechanism composed of a low friction and low backlash 1:10 Alpha gearbox, a 60 mm diameter pulley and timing belt. The linear velocity of the carriage is calculated using the signal of angular velocity provided by the BLH75 motor drive and considering the global transmission ratio.

This experimental setup is equipped with appropriate hardware signal conditioning and electronics systems needed to perform security tasks and data acquisition. It uses a Pentium 4 Personal Computer with a PCI DAS1602-12 I/O card and a PCI-QUAD04 encoder card from Measurement Computing.

III. GLOBAL CONTROL ARCHITECTURE

The control system is implemented on a Personal Computer and runs on a real-time environment provided by the MATLAB/Simulink software package. This application is very powerful since it integrates a real-time module with simulation and development environments, in a very user friendly manner. It is then possible to implement a really open control system, where control algorithms can be simulated, implemented and tested. Furthermore, as there is the additional request to provide a remote interaction with the system, the MATLAB/Simulink application not only offers Web integration but also allows the integration of other standard applications. Thus, the control architecture of the inverted pendulum was organized into two modules. In the first module, it was developed a control application with a graphical interface to operate locally the inverted pendulum. The local user interacts with the system through this application, allowing the command and monitoring of the system. Detailed information on the implemented application is presented in section IV. The second module refers to the remote operation and monitoring of the system, being done by a remote user on a remote computer connected to the local computer through the Web. The developed application uses a server/client architecture and will be referred in more detail in section V.

IV. LOCAL CONTROL APPLICATION

The local control application implements a graphical interface that enables the user to command and monitor the system. This application was developed in the .Net Framework, using the C# language and communicates
Remote Web operation of an inverted pendulum

The developed control application includes a state machine implemented with the Stateflow toolbox that includes all possible modes of operation of the pendulum, as well as all possible transitions between them (see Fig. 3). The state machine inputs are the signals coming from the pendulum and carriage, from the local graphical interface and from the interface parameters given by the remote user (throughout the server application that will be referred later). With this data, the Stateflow block implements the appropriated control strategy, swing-up strategy or stabilization control, deciding on the type of trajectory the carriage must follow.

The input signals of the Stateflow block can trigger the transitions between the different states and control strategies. As an example, the switch between the swing-up strategies to trajectory control occurs when the angular position of the pendulum reaches the value that was defined by the user, when the control strategy was configured.

V. REMOTE WEB CONTROL APPLICATION

To implement the remote Web control, it was adopted a server/client architecture (Fig. 4). The server is implemented on the local computer, running under Windows XP, and is responsible for communicating between the client side and the local control application. This Web server was developed from scratch using Visual Studio, .NET and the C# language. The programming C# language was adopted due to its inherent simplicity, providing a high efficiency and enabling the use of “background workers” to generate and manage events required by the application. At the same time, the .NET platform is able to offer an easy integration of different applications. It was possible to integrate the server with the local control application developed in MATLAB/Simulink, making use of the MATLAB Application Type Library available within MATLAB version R2006a. Fig. 5 shows the structure of the server application and the flow of information.

The server application has other functionalities accessible through menus such as the one where it is possible to verify if there is any client connected and see what type of commands are being sent, apart from starting the server and disconnecting the client session (see Fig. 6). The global working model of the server application is presented in the flowchart shown in Fig. 7.
Remote Web operation of an inverted pendulum

The client communication module of the server includes a socket module to establish the connection, an event handler and a client management module. When a client intends to connect, his respective request generates an event that calls the client management module. The objective of this module is to assure that there is only one client that has the total control of the physical system and to report that fact to other incoming requests that may appear. After a valid connection, the server waits for the client command orders, implemented throughout a programmed protocol. The implementation of these command orders was made using the MATLAB Application Type Library. It is also through this process that the data of the pendulum and carriage behaviour is sent to the client at a rate of 1 Hz.

The client application enables the user to interact with the system through the server. The interaction of the remote user through the Web requires an application that can run under a Web browser with the flexibility to receive and send data dynamically. These requests led to the use of Java language to program an Applet to run continuously on a Web browser. The remote user is able to define the control strategy (among a predefined set) and to adjust the control parameters as well as seeing the graphical response of the physical system. Thus, the client application consists of a communication module and a graphical interface (see Fig. 8).

The client application includes, in a manner similar to the server application, a socket communication module used to send/receive the data. Strip chart creation is available to appreciate the behaviour of the pendulum and carriage system. The data is received and a polling process is implemented to read it, according to the communication protocol that was created. The data is shown on two strip charts. One strip chart shows the position of the carriage and the respective reference command and the other strip chart shows the angle between the pendulum and a vertical direction.

Another main module of the client application is the graphical interface (Fig. 9). This interface allows the user to input different control parameters and to choose the control strategy to implement. Three different strategies where implemented:

- Swing-up strategy;
- Control type for the trajectory following of the carriage;
- Reference command for the movement of the carriage.

The client application is made available by a HTTP server (Apache) installed in the server computer.
Remote Web operation of an inverted pendulum

VI. CONCLUSIONS

This paper reports the development of an experimental test rig of an inverted pendulum and the development of the control system to be able to operate the pendulum locally and remotely, through the Web.

The MATLAB/Simulink application was used, not only to implement the control strategies but also to provide the real-time environment necessary to implement the control of the system. The proposed global control architecture and the developed control software made possible the remote operation of the inverted pendulum. This software platform proved to be very versatile. It was possible not only to develop and simulate different control strategies, but also to integrate a server/client application to enable the remote control.

The use of this control platform has also the possibility to integrate virtual reality systems. Building of a virtual model of the inverted pendulum, using the already available 3D model of the system, and applying the developed control strategies is envisaged.

ACKNOWLEDGMENT

The authors would like to recognize the work carried out by their students, in particular Luis Pena and Miguel Gonçalves.

This work was supported in part by “Universidade do Porto – Caixa Geral de Depósitos: Investigação Científica na Pré-Graduação”.

REFERENCES


AUTHORS

Manuel Rodrigues Quintas is with Universidade do Porto, Faculdade de Engenharia, Departamento de Engenharia Mecânica e Gestão Industrial, Rua Dr. Roberto Frias, 4200-465 Porto, Portugal (mrq@fe.up.pt).

António Mendes Lopes is with Universidade do Porto, Faculdade de Engenharia, Departamento de Engenharia Mecânica e Gestão Industrial, Rua Dr. Roberto Frias, 4200-465 Porto, Portugal (aml@fe.up.pt).

Carlos Moreira da Silva is with Universidade do Porto, Faculdade de Engenharia, Departamento de Engenharia Mecânica e Gestão Industrial, Rua Dr. Roberto Frias, 4200-465 Porto, Portugal (csilva@fe.up.pt).

Paulo Abreu is with Universidade do Porto, Faculdade de Engenharia, Departamento de Engenharia Mecânica e Gestão Industrial, Rua Dr. Roberto Frias, 4200-465 Porto, Portugal (pabreu@fe.up.pt).

Diego Sá is a postgraduate student in the MSc Automation, Instrumentation and Control, Universidade do Porto, Faculdade de Engenharia, Departamento de Engenharia Mecânica e Gestão Industrial, Rua Dr. Roberto Frias, 4200-465 Porto, Portugal (mai06004@fe.up.pt).