Chapter 18
Matlab RTW–Based Internet Accessible Remote Laboratory for Teaching Robot Control

Zdenko Kovačić
University of Zagreb, Croatia

Davor Jerbić
University of Zagreb, Croatia

Vedran Vojvoda
University of Zagreb, Croatia

Siniša Dujmović
University of Zagreb, Croatia

ABSTRACT
This chapter describes the use of Matlab Real Time Workshop (RTW) for implementing an Internet Accessible Laboratory (IAL) for teaching robot control. The IAL architecture consists of three key components - IAL Web Application, IAL database, and a set of robot control schemes prepared for students’ laboratory curriculum that are running in Matlab RTW. The IAL management system supports multilingual access and enables easy addition of new users, new robotic systems, and new laboratory exercises related to robot control. The IAL functionality is demonstrated with the example of controlling a four degrees of freedom SCARA robot.

INTRODUCTION
The aim of remotely controlled robotic systems is to execute a task given to a robot without physical contact of a robot operator with the environment the robot works in. There are numerous applications of such robot systems, some of them dating from early days of robotics, including manipulations in radioactive surroundings, explosive and poisoned atmospheres, and other locations unsafe for human beings (Sheridan, 1992). Nowadays applications of remotely controlled robotic systems include, for example, remotely operated robots for surgical operations, home assistant robots,
nursery robots and most recently, entertainment and therapeutic robots (Cao et al., 1995), (Tachi, 1998), (Goldberg et al., 1995), (Preusche, Ortmaier, & Hirzinger, 2002).

A large number of remote robot control applications have evolved, thanks to the rapid development of Internet technology (Taylor & Dalton, 2000), (Saucy & Mondada, 2000), (Barney & Kenn, 2000), (Marin et al., 2005). The use of the Internet for remote control has caused a number of new problems, mostly due to random communication time delays depending on Internet traffic and inherent latency of the network that affects the stability of such a control (Xi & Tarn, 2000), (Fiorini & Oboe, 1997), (Liu et al., 2006).

With the development of advanced e-learning systems, more and more attention has been paid to remotely operated robot systems used in education (Safaric et al., 2005), (Jara, Candelas, & Torres, 2008), (Tzafestas, 2009). The purchase of rather expensive robotic equipment for educational purposes has always been a problem. Eventually, this becomes a serious obstacle for effective organization of laboratory assignments that involve a large number of students enrolled in robotics courses (Minamide et al., 2007). By adopting an Internet-based organization of laboratory activities, around-the-clock access to the available robotic equipment becomes possible.

The main deficiency of working in the remote robot control lab is the absence of user’s audiovisual and physical contact with the controlled robot. This can be easily overcome by inclusion of a live video-stream from a remote supervisory camera or a 3D visualization of robot motion. For example, a 3D visualization can be based on the use of virtual 3D models and feedback information obtained from the sensors mounted on the operating robot (Genter, Munk & Kovacic, 2004), (Draganjac et al., 2008). In order to follow the progress of students’ work and simultaneously control the use of laboratory resources in a structured way, a remote e-learning laboratory system must have an infrastructure that can add new users, change user access privileges and control logging of user actions. All these elements should be integrated during the laboratory setup (Zubia et al., 2007).

Using software packages for modeling, simulation, and optimization of control systems has become a regular engineering practice. An added possibility to generate real-time executable code directly from simulation models enabled shorter development times and faster validation of new control solutions. Due to Matlab-Simulink being one of the most popular simulation software packages (Matlab, 2010), many remote robot control solutions were built using Matlab and its various toolboxes (Casini, Prattichizzo, & Vicino, 2009), (Turan, Bogosyan, & Gokasan, 2006). One particularly interesting is the Real Time Workshop (RTW) Toolbox, which can interpret a Matlab-Simulink control scheme and generate the code for “real time” execution. This has been a motive for developing remote robot control laboratories using Matlab RTW (Castellanos et al., 2006), (Kovacic et al., 2007).

Matlab RTW has been used as the backbone of the Internet Accessible Laboratory (IAL) presented herein. Matlab RTW is connected to the Internet by the Matlab web server. Also, a client function block for Simulink was built as an interface between the Matlab RTW and target robot control hardware. The IAL management system controls the access of potential users. Based on the first come-first serve principle, users reserve a time slot and a desired laboratory exercise. The developed IAL management system enables easy addition of new users and new exercises. Being built as a multilingual system, the procedure for adding a new language to the IAL is very simple.

Taking into account problems that network latency and the lack of proper synchronization of Matlab RTW and the targeted control system may cause, it has been decided not to allow any user’s interaction with the robot control system during execution of real-time robot control. Also, in order to prevent possible synchronization losses between the Matlab RTW and the targeted control