

A Web-Based Laboratory on Control of a Two-Degrees-of-Freedom Helicopter*

C. C. KO, BEN M. CHEN, JIANPING CHEN, JING ZHANG and K. C. TAN

Department of Electrical and Computer Engineering, The National University of Singapore, Singapore 117576. E-mail: bmchen@nus.edu.sg

Web-based laboratories are systems that enable students to conduct actual laboratory experiments anywhere at anytime via the Internet. The emergence of the web-based laboratories is an innovative advancement in engineering education and indicates the trend of distant learning. In this paper, we aim to exploit the philosophy and development of designing an efficient, secure and convenient web-based laboratory for a two-degrees-of-freedom (2DOF) helicopter. In particular, we will discuss issues related to the hardware and software structures, network communications, audio and video feedback as well as camera control systems of the web-based laboratory. We will also present the design and implementation of a series of linear and nonlinear controllers for the 2DOF helicopter through the web-based laboratory. The physical structure of the 2DOF helicopter makes it an ideal platform for implementing and evaluating control strategies such as proportional-integral-derivative (PID) control, fuzzy control and state space feedback control. The system is appropriate for all levels of university education and research. The web-based laboratory is currently being utilized in teaching both undergraduate and postgraduate courses in the Department of Electrical and Computer Engineering at the National University of Singapore. The system is particularly beneficial to part-time students, who are unable to access to the laboratory facilities during normal operating hours. Feedback and comments from the students are very positive.

INTRODUCTION

SPURRED by development in computer science and network technology, the use of the Internet has been expanding exponentially and is now extensively utilized as a connectivity and reference tool for commercial, personal and educational purposes. Especially in the education field, the Internet provides many ways to enhance learning and expand educational opportunities for students. Distance education and non-traditional classrooms reach more students with specialized instruction and self-paced learning. The usual methods that integrate the Internet into education can be divided into three categories [1]:

1. Develop a course website to centrally house various online functions and facilitate course management.
2. Create a remote laboratory in which multimedia animation or simulation to replace physical experiments are provided.
3. Develop a web-based laboratory that enables students to set up the parameters for running experiments from a remote location.

Successful examples with the first two capabilities can be found in [2–5]. In [3], the authors presented a user-centered web page construction and maintenance model to develop web-based course materials at the University of Minnesota College of Education and Human Development (CEHD). In

[4], Pascoe not only made use of the Internet for distributing course materials, but also developed several methods for enabling students to interact with the course website to enhance their learning. The methods include interactive exercises, facilities for students to annotate course notes, improved graphics and automatic feedback tailored to specific student needs. In [5], Rosenblum, et al., developed web-based communication tools through a Common Gateway Interface (CGI) program that provides instructors with private course discussion areas. With an intuitive chat interface, these areas allow instructors to give students a platform-independent ability to communicate in as many groups as is needed by the class.

However, the above web-based laboratories only provide students theoretical and/or simulation material. It is commonly recognized that efficient learning requires a mixture of theoretical and practical exercises. For the sake of understanding how to apply theoretical knowledge to the real world, practical exercises become particularly important. Simulations can only provide theoretical exercises and in many cases they are just approximations of the real world. Several web-based laboratories that enable students to set up parameters for running experiments from a remote location have been developed [6–9].

In this paper, we design and implement a series of controllers for a plant through a web-based laboratory at the National University of Singapore. The web-based laboratory is thought to be an innovative platform for implementing and eval-

* Accepted 26 July 2005.



Fig. 1. The 2DOF helicopter apparatus.

uating control strategies. With the web-based laboratory, users can verify their control algorithms on a physical plant without the need to acquire the plant permanently. The user logs on and posts control parameters from a remote client to a web server that is connected to the plant. The server receives the parameters and uses these in an appropriate controller to control the physical plant. There are also a camera and a microphone next to the plant to capture the response of the plant in real time. The captured data are then sent to the user so that the user has a realistic feel of applying his algorithm to the real plant.

We choose a 2-degrees-of-freedom flight simulator [10], a 2DOF helicopter, as the control plant. As shown in Fig. 1, the 2DOF helicopter consists of a model mounted on a fixed base. The model

has two propellers driven by DC motors that are mounted at the two ends of a rectangular frame.

The motors are mounted at 90° such that one causes pitch and the other causes yaw of the helicopter. The helicopter frame is free to rotate on a vertical base equipped with a slip ring. Electrical signals to and from the helicopter are channeled through the slip ring to eliminate tangled wires, reduce friction and allow for unlimited and unhindered yaw. The coupling between the pitch and yaw motor torques results in a coupled 2-input/2-output system. The physical structure makes the 2DOF helicopter appropriate for all levels of university education and research. It is an ideal platform for implementing and evaluating feedback strategies such as PID control, LQ control, fuzzy control, H_2 control and H_∞ control.

STRUCTURE OF WEB-BASED LABORATORY

We present in this section the hardware and software structures as well as the network configuration of the web-based laboratory for the 2DOF helicopter.

Hardware structure

The overall structure of the web-based laboratory is shown in Fig. 2. A client PC accesses the HTTP (Hypertext Transfer Protocol) server that hosts the website for conducting the experiment. A PC is configured as the local instrument controller and is connected to the HTTP server. The local PC is connected to the helicopter through a DAQ (Data Acquisition) card and implements local control of the physical helicopter by supplying

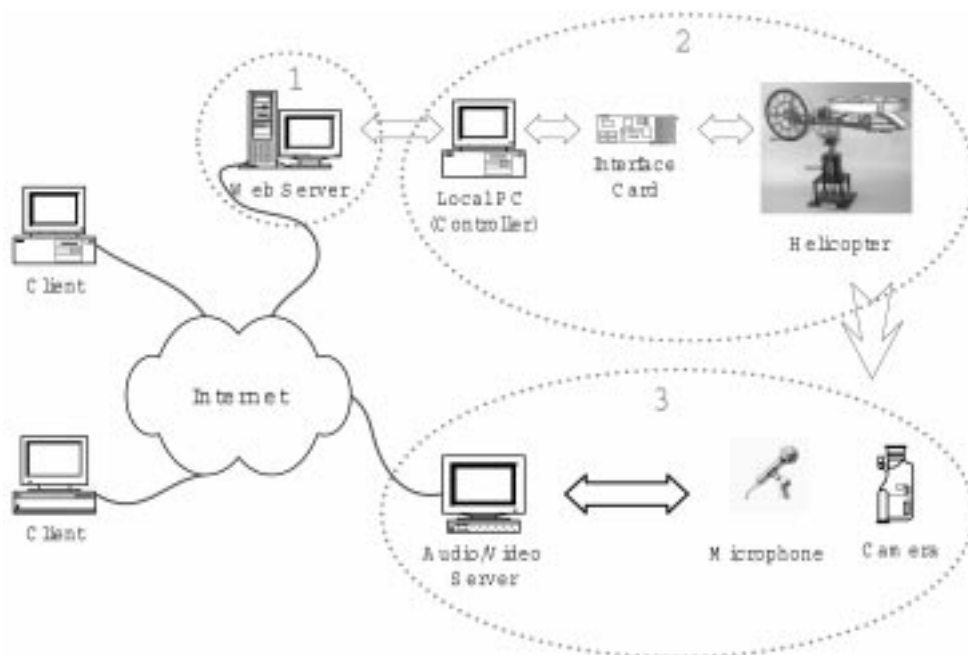


Fig. 2. Hardware structure of the web-based laboratory.

two voltages to the helicopter. A camera connecting to a PC running Microsoft NetMeeting and serving as the video/audio server enables the provision of visual feedback to the user on how the helicopter moves. A microphone connected to the same PC captures the sound of the helicopter motor and provides audio feedback to the user. The pan and tilt of the camera as well as its zoom setting are controlled by a circuit board supplied by American Dynamics. This circuit is also connected to the instrument controller. The live image data captured by the camera and the sound captured by the microphone are sent to the client through a video conferencing session. In multicast mode, the web-based laboratory allows one user to conduct the experiment at a time, whereas several users can view what is happening in the web-based laboratory.

TCP for client server communication

CGI (Common Gateway Interface) and TCP (Transmission Control Protocol) are two popular methods for implementing client-server communication, as illustrated in Fig. 3. CGI programs running on a web server for sending and receiving parameters from the users' web browsers have become an Internet standard. The client uses an Internet browser to connect to the web server, view the web page, send a CGI request to the server and finally receive response from the server. Since CGI connection is based on HTTP, which is a per-session basis protocol, the connection between the client and the server is closed once the client receives the response from the server.

In contrast to a CGI connection, a TCP connection established by a client with a server remains in place until the client closes the connection. Since the establishment of a TCP connection implies that the connection will be in place until the client closes it, TCP is the ideal communication protocol for implementing web-based instrument control that requires frequent parameter adjustments. CGI, on the other hand, is not so suitable for this purpose due to its operation on a per-session basis. Figure 4 illustrates the communication between a client and a server using TCP.

Communication between the client and server can be set up conveniently using Java. Java programs are compiled to platform independent codes, called bytecodes, of a hypothetical machine called the JVM (Java Virtual Machine). These bytecode programs, often called applet, can be included in a web page in the same way as an image. In addition, they can be viewed and downloaded by a Java-enabled browser to be executed by the browser JVM. Using Java, the connection between the client and the server side can be set up through sockets. When the user downloads a Java applet from the web server, the applet (running on the client machine) represents the client side of the connection. Java was preferred to other programming languages for implementing client-server communication due to its platform independent nature and extensive network programming support.

Audio and video feedback

A striking feature of the implementation is the use of video conferencing and multicast schemes to



Fig. 3. A typical CGI connection.

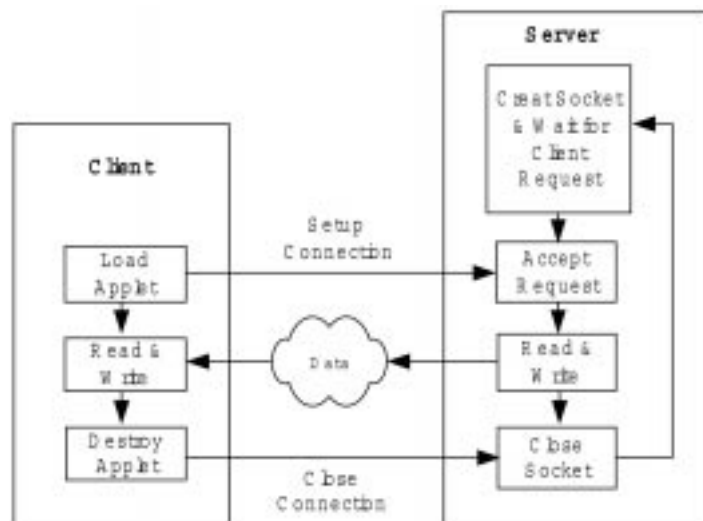


Fig. 4. A typical TCP connection.

provide the user with audio-visual feedback. Since the NUS campus network, NUSNET-III, provides the user with bandwidth comparable to a T1 connection, bandwidth is not a constraint for the implementation of remote laboratory experiment within the campus and video conferencing was adopted to provide live video to the client. Hence, Windows NetMeeting 3.0 was chosen. NetMeeting uses the H.323 standard, which is comprised of ITU (International Telecommunications Union) approved protocols for audio, video, and data conferencing over TCP/IP networks. The standard is compatible with H.261 and H.263 video codecs.

As NetMeeting 3.0 also supports the transfer of audio, a microphone connected to the instrument controller is used to capture the sound made by the helicopter motor. The latter can then be heard at the client side by attaching speakers to the client computer. This feature is intended to provide the user a feel which is as realistic as possible while performing the experiment. The implementation uses an ActiveX control to embed the live audio and video on web pages. This control is part of the Windows NetMeeting Resource Kit, and is invoked and controlled using VBScript, a scripting language developed by Microsoft Corporation. When the user activates ActiveX and calls the video server (explicitly by specifying the IP (Internet Protocol) of the video server), a call is placed. The NetMeeting program on the video server is configured to receive calls immediately without any user intervention on the server side to accept the call. The program is also configured to send out live audio and video at the start of each call. The audio and video at the

client side can be stopped in a similar manner through ending the call or clicking the 'Stop A/V' button.

Camera control

Another special feature of the implementation is the camera's pan, tilt and zoom control. The camera is mounted on a pan and tilt unit, which can be controlled through a circuit board. The board is in turn connected through RS-232 to the controller PC's serial port. The web-based control is implemented by continuously running a camera control program on the machine housing the HTTP server to receive command strings from the client (through a TCP channel) for the local control of the circuit board.

At the client end, a Java applet sets up the sockets for TCP communication with the server. When the user clicks a button intended for camera control, a procedure in the applet that generates the corresponding command string and writes it to the TCP connection is invoked.

Software structure

Figure 5 shows the software structure of the web-based laboratory. The PC running the LabView HTTP server also runs the LabView G language programs that set up the server side of the TCP communication with the client. As shown in Fig. 5, there are several Java applets running on the client for handling the necessary communication with the web server and plotting the response curve using the data obtained from the server. Also there is an applet communicating with a server program by sending it command strings for camera zoom and pan/tilt control. An ActiveX

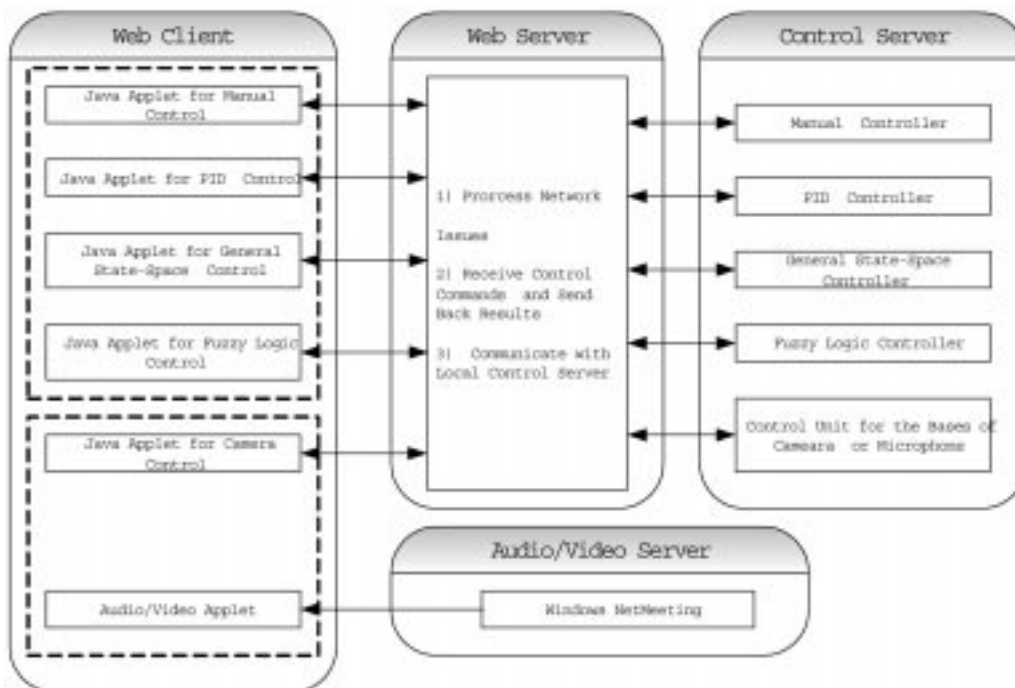


Fig. 5. Software structure of the web-based laboratory.

control executes on the client and receives audio and video data from the video server.

On the web server, there are several programs handling communication with the client and receiving the controller parameters and the reference input (set point) values input by the user. These parameters are passed on to the control server (coded in LabView G) which implement the local control of the helicopter. The program for controlling the mounting unit of the camera is coded in Visual Basic and also resides on the same machine with the web server.

WEB INTERFACES AND CONTROLLERS FOR THE EXPERIMENTS

The helicopter model

In this subsection, we present the model of the helicopter with its schematic diagram as shown in Fig. 6. The pitch propeller is driven by a DC motor whose speed is controlled via input voltage u_{pitch} . The rotation of the propeller results in a force that acts normal to the body at a distance R_p from the pitch axis. The rotation however also causes a load torque T_p on the motor shaft which is in turn seen at the yaw axis. Thus rotating the pitch propeller not only causes motion on the pitch axis but also on the yaw axis. Similarly, the yaw motor controlled by the input u_{yaw} causes a force F_y to act on the body at a distance R_y from the yaw axis and a torque T_y which is experienced about the pitch axis. Hence, it is very difficult to derive a

complete mathematical model from the physical structure of the helicopter.

We utilize the frequency response identification method [11] to model the helicopter. The method is applicable to minimum phase systems. By observing its physical properties, we find that the helicopter meets the requirements to be a minimum phase system. We will now formulate the helicopter system as a 2×2 multi-variable transfer function:

$$Y(s) = G(s)U(s), \tag{1}$$

where

$$Y(s) = \begin{bmatrix} y_1(s) \\ y_2(s) \end{bmatrix}, \quad U(s) = \begin{bmatrix} u_1(s) \\ u_2(s) \end{bmatrix},$$

$$G(s) = \begin{bmatrix} g_{11}(s) & g_{12}(s) \\ g_{21}(s) & g_{22}(s) \end{bmatrix}, \tag{2}$$

$y_1(s)$ and $y_2(s)$ are Laplace transforms of the pitch output $y_{pitch}(t)$ (in degrees) and yaw output $y_{yaw}(t)$ (in degrees), and $u_1(s)$ and $u_2(s)$ are Laplace transforms of the control signals for the pitch $u_{pitch}(t)$ (in volts) and yaw $u_{yaw}(t)$ (in volts), respectively. The working ranges of $u_{pitch}(t)$ and $u_{yaw}(t)$ are from 0 to 2.5 V and from -2.5 volts to 2.5 V, respectively. Using the algorithm of [11] and the data measured from the real system, we obtained the following model for the helicopter:

$$G(s) = \begin{bmatrix} \frac{3.9882s+145.2350}{s^2+3.1273s+10.5480} & \frac{5.7188}{s+0.3529} \\ \frac{-80}{s} & \frac{260}{s} \end{bmatrix}. \tag{3}$$

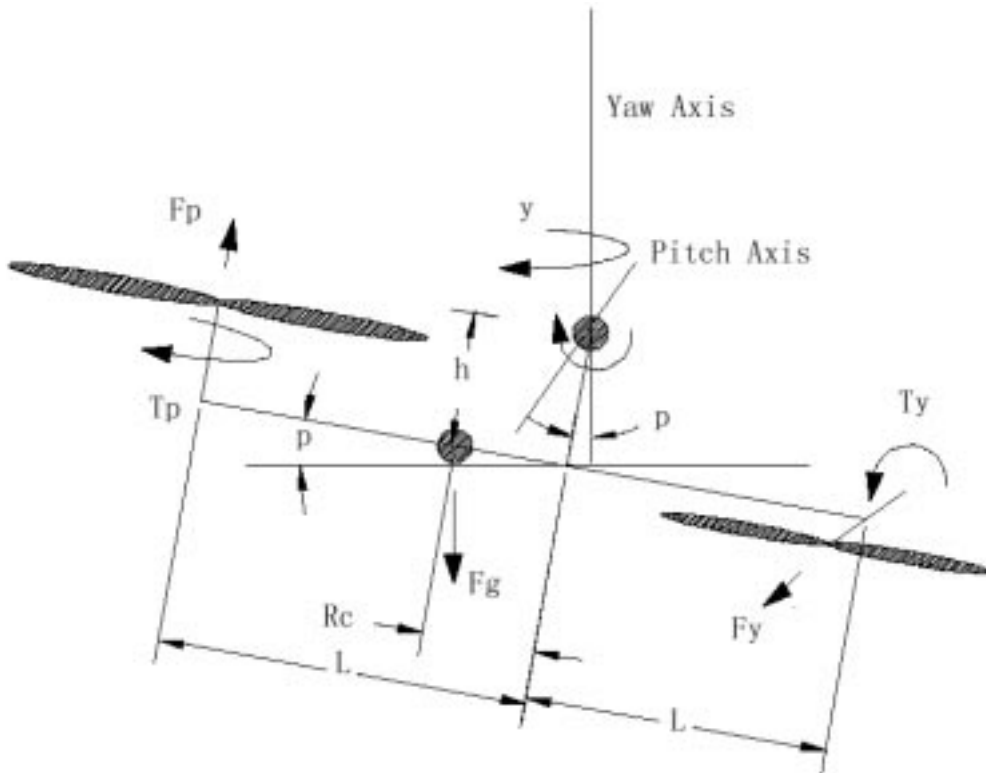


Fig. 6. Schematic diagram of the helicopter.

Figure 7 shows the frequency responses of the actual and identified models of the helicopter. It can be seen that the identified model and measured data match quite well from 0 to 10 rad/sec. This matched band is wide enough for us to design a working controller. For future use, the state space realization of the system is given by:

$$\begin{cases} \dot{x} = Ax + Bu, \\ y = Cx + Du, \end{cases} \quad (4)$$

where

$$A = \begin{bmatrix} -1.5637 & 2.8466 & 0 & 0 \\ -2.8466 & -1.5637 & 0 & 0 \\ 0 & 0 & -0.3529 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix},$$

$$B = \begin{bmatrix} -12.1326 & 0 \\ 0 & 0 \\ 0 & 2 \\ -5 & 16.25 \end{bmatrix}, \quad (5)$$

and

$$C = \begin{bmatrix} -0.3287 & 4.0247 & 2.8593 & 0 \\ 0 & 0 & 0 & 16 \end{bmatrix},$$

$$D = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}. \quad (6)$$

Design and implementation of controllers

There are four different control schemes which have been implemented in the web-based laboratory setup for the helicopter. These are: PID control, decoupled PID control, state-space control and fuzzy logic control. In the following, we will briefly describe how these are applied to the helicopter.

PID control. A typical PID controller can be specified by:

$$u = K_p e + K_i \int edt + K_d \dot{e}, \quad (7)$$

where $e = r - y$ is tracking error, r is the reference, K_p , K_i and K_d are proportional, integral, and

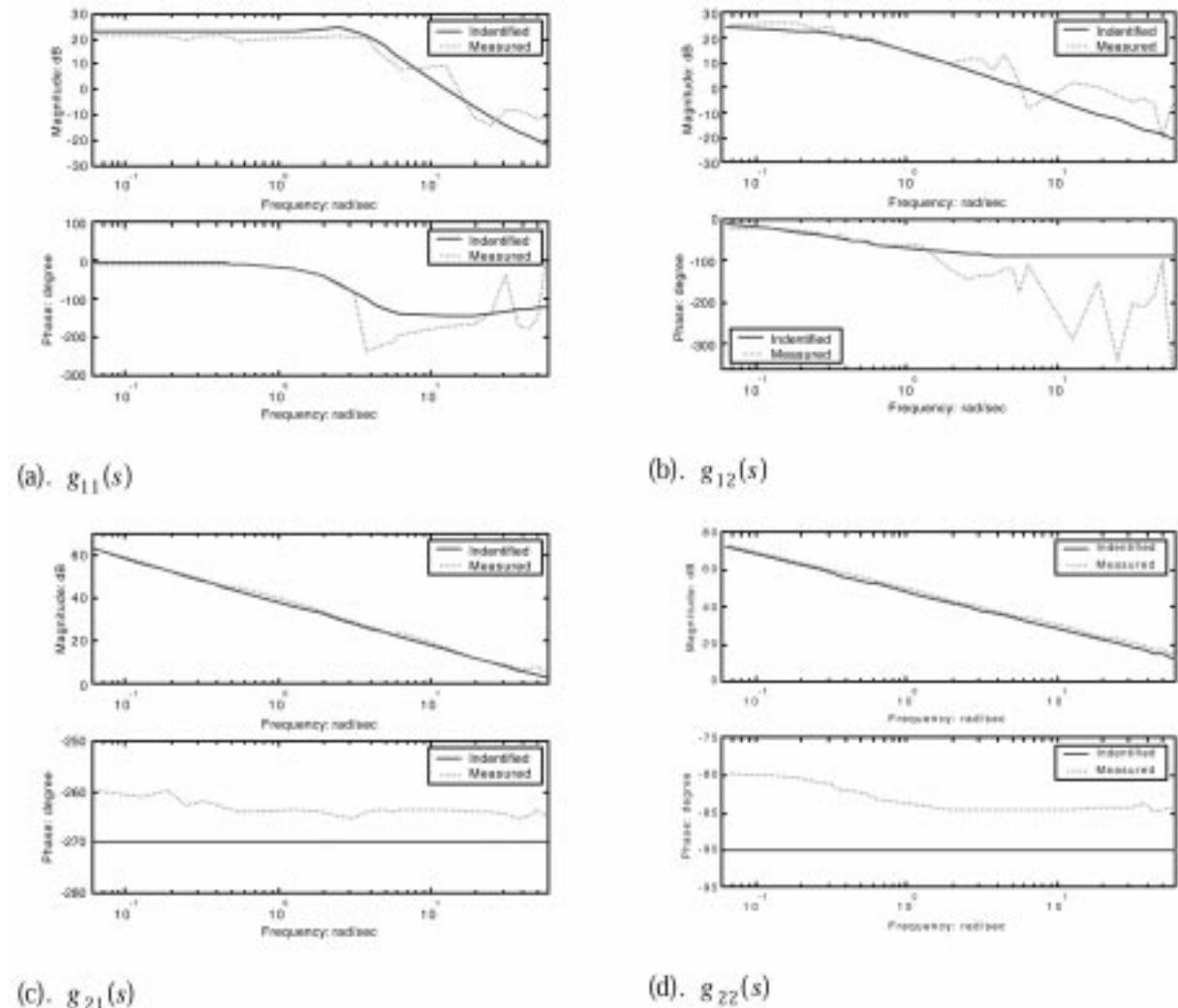


Fig. 7. Frequency responses of the actual and identified.

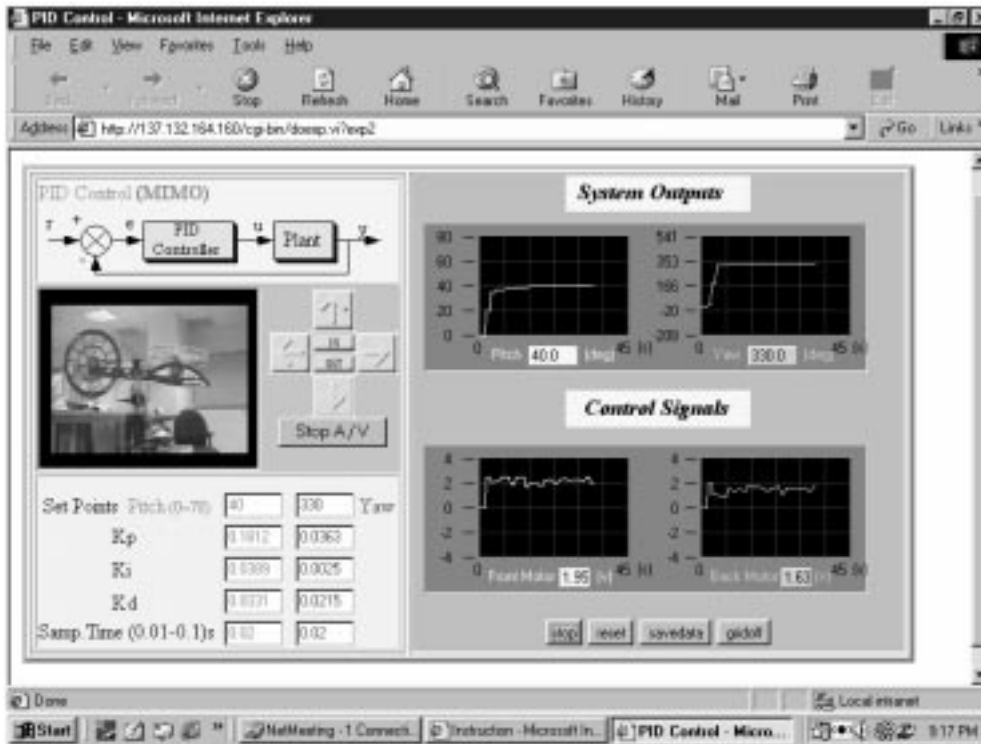


Fig. 8. Interface of PID control in the web-based laboratory.

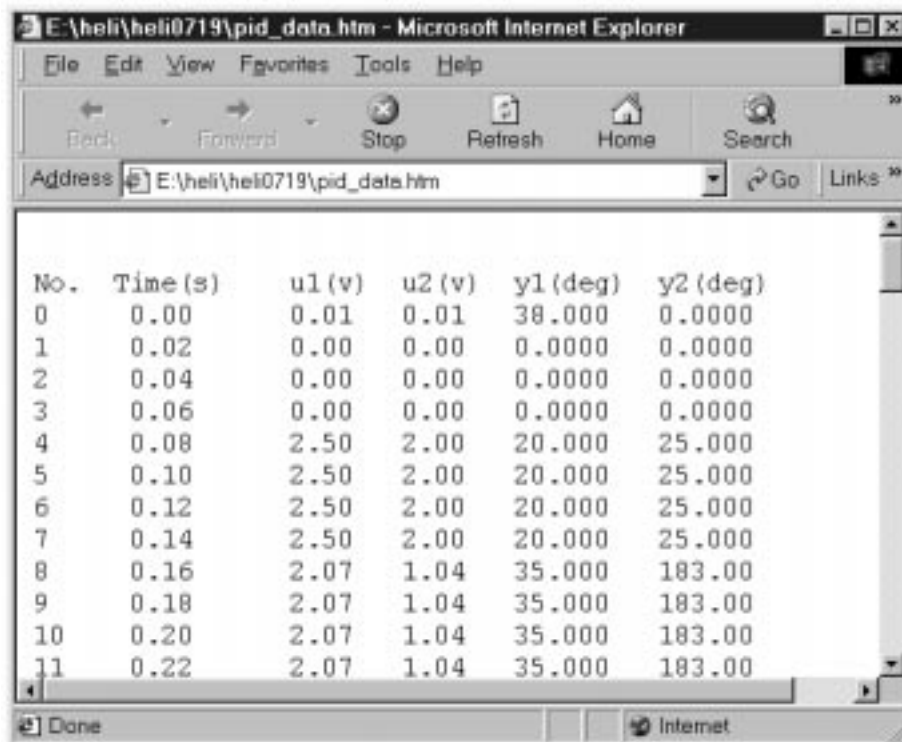


Fig. 9. Experiment results downloadable to the user.

derivative gains, respectively. Using the method in [12] together with online manual tuning, we obtain two controllers for the pitch and yaw channels of the plant: $K_{p1} = 0.1812$, $K_{i1} = 0.0389$, $K_{d1} = 0.0331$, and $K_{p2} = 0.0363$, $K_{i2} = 0.0025$, $K_{d2} = 0.0215$, respectively. Since the system is PC-based, the controller has to be discretized before applying to

the real system. The sampling time for the system is 0.02 s. Figure 8 depicts the interface of the web-based laboratory under PID control.

In situations where the network is congested or where the available network bandwidth is very small, the result curves of the experiment may not be smoothly displayed. To compensate for this

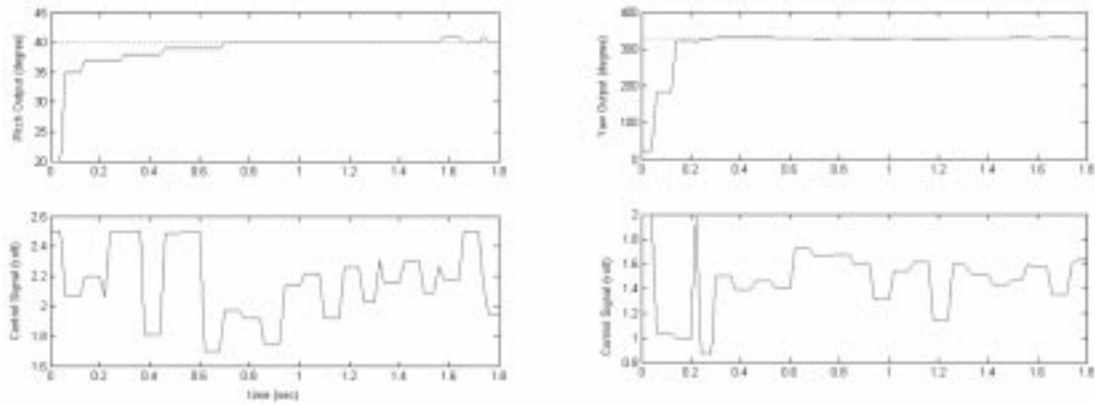


Fig. 10. Experiment results processed by MATLAB.

deficiency, the web-based laboratory includes the capability for the user to download all the experiment results. Figure 9 shows the format of the downloaded data file, while Fig. 10 depicts experimental results after processing by MATLAB.

Decoupled PID control. From the step responses, we find that, with pitch and yaw coupled, it is not easy to design high quality PID controllers. On the other hand, we find that both:

$$\frac{g_{21}(s)}{g_{22}(s)} = -0.3077 \quad (8)$$

and

$$\begin{aligned} \frac{g_{12}(s)}{g_{11}(s)} &= \frac{5.7186(s^2 + 3.1273s + 10.5480)}{(s + 0.3529)(3.9882s + 145.2350)} \\ &= \frac{1.4339(s^2 + 3.1273s + 10.5480)}{(s + 0.3529)(s + 36.4126)} \end{aligned} \quad (9)$$

are realizable. Hence, the two subsystems can be decoupled.

Figure 11 shows the block diagram of a decoupled PID controller. According to the figure, after feed-forward compensation, we have:

$$\begin{aligned} y_1(s) &= g_{11}(s)u_1(s) + g_{12}(s)u_2(s) \\ &= g_{11}(s)(v_1(s) - \frac{g_{12}(s)}{g_{11}(s)}u_2(s)) + g_{12}(s)u_2(s) \\ &= g_{11}(s)v_1(s) \end{aligned} \quad (10)$$

and

$$\begin{aligned} y_2(s) &= g_{21}(s)u_1(s) + g_{22}(s)u_2(s) \\ &= g_{21}(s)u_1(s) + g_{22}(s)(v_2(s) - \frac{g_{21}(s)}{g_{22}(s)}u_1(s)) \\ &= g_{22}(s)v_2(s), \end{aligned} \quad (11)$$

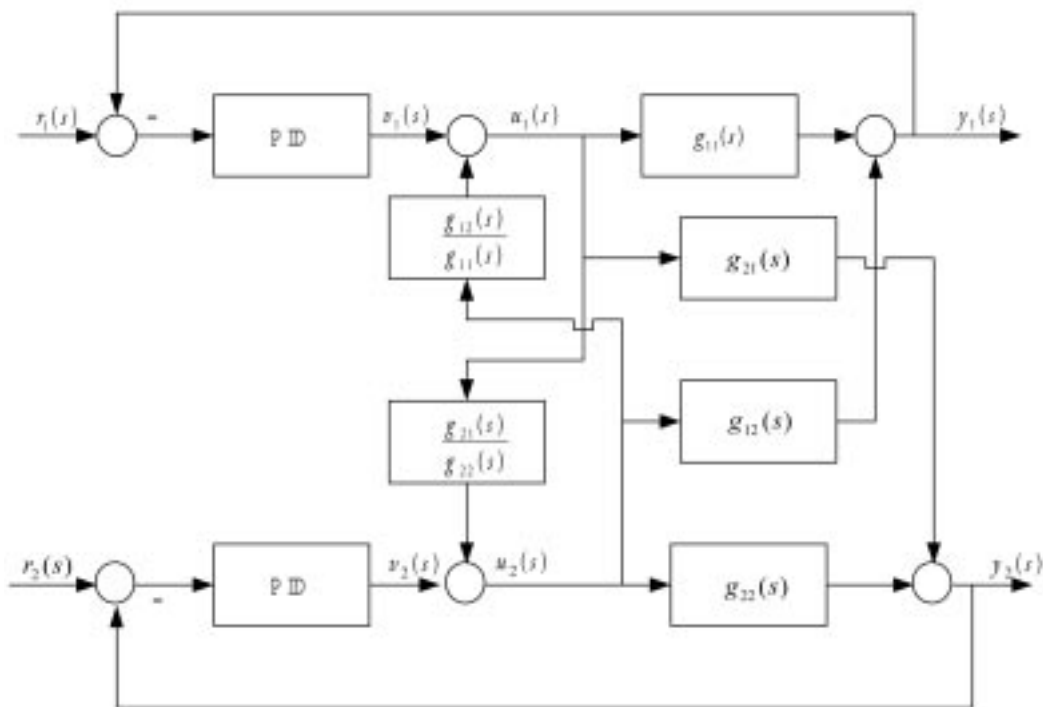


Fig. 11. Block diagram of the decoupled PID controllers.

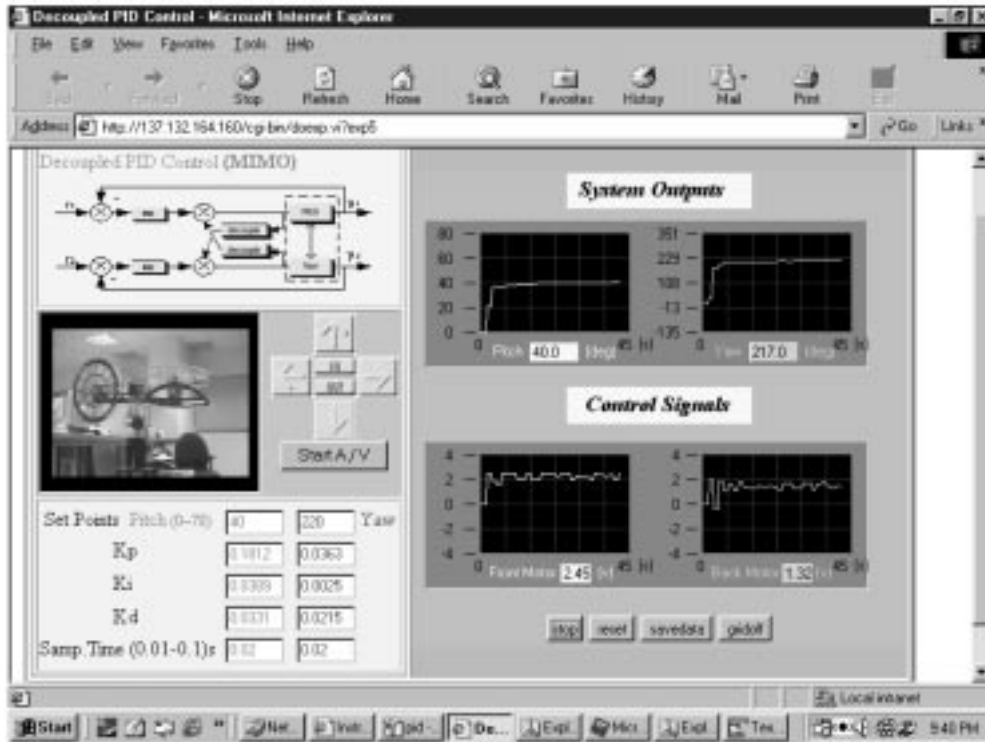


Fig. 12. Interface of decoupled PID control in the web-based laboratory.

which imply the possibility of two separate PID controllers for pitch and yaw, respectively.

Combining the method in [12] and online tuning, we obtain two decoupled PID controllers given by $K_{p1} = 0.1812$, $K_{i1} = 0.0389$, $K_{d1} = 0.0331$ and $K_{p2} = 0.0363$, $K_{i2} = 0.0025$, and $K_{d2} = 0.0215$. Figure 12 depicts the interface of the web-based laboratory with the decoupled PID controller.

General state space control. Figure 13 shows the block diagram for the helicopter when a general state space controller is used. Here, r is the reference input, y is the measured output from the helicopter, and u is the control input. The state space model of the controller is given by:

$$\begin{cases} \dot{x}_c = A_c x_c + B_c y + G_c r, \\ u = C_c x_c + D_c y + H_c r \end{cases} \quad (12)$$

where x_c is the state of the controller, A_c , B_c , G_c , C_c , D_c , and H_c are constant matrices with appropriate dimensions. We note that all linear controllers, such as Linear Quadratic Gaussian (LQG), H_2 and H_∞ optimal control, can be transformed into the format of (12). The following controller data are obtained from an LQ control formulation:

$$A_c = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & -100 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & -100 \end{bmatrix},$$

$$B_c = \begin{bmatrix} 0 & 0 \\ -100 & 0 \\ -10000 & 0 \\ 0 & 0 \\ 0 & -100 \\ 0 & -10000 \end{bmatrix},$$

$$G_c = \begin{bmatrix} 0 & 0 \\ 100 & 0 \\ 10000 & 0 \\ 0 & 0 \\ 0 & 100 \\ 0 & 10000 \end{bmatrix}, \quad (13)$$

$$C_c = \begin{bmatrix} 0.0389 & 0.1812 & -0.0331 & -0.0011 & -0.0150 & 0.0009 \\ 0.0007 & 0.0035 & -0.0023 & 0.0025 & 0.0363 & -0.0215 \end{bmatrix}, \quad (14)$$

and

$$D_c = \begin{bmatrix} -3.31 & 0.09 \\ -0.23 & -2.15 \end{bmatrix}, \quad H_c = \begin{bmatrix} 3.31 & -0.09 \\ 0.23 & 2.15 \end{bmatrix}. \quad (15)$$

The interfaces of the web-based laboratory using general state space control are given in Figs 14 and 15.

Knowledge-based fuzzy control

The helicopter system can also be controlled by a FKBC (Fuzzy Knowledge-Based Controller).

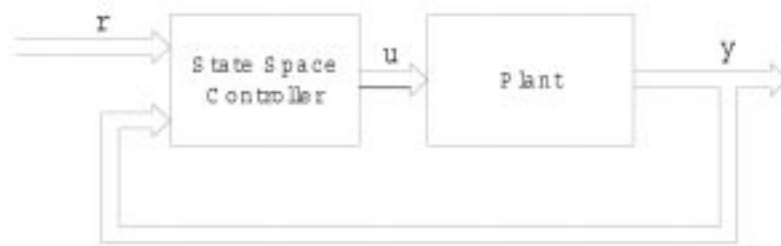


Fig. 13. Block diagram of control system with state space controller.



Fig. 14. Interface of state space control in the web-based laboratory.

Figure 16 shows the system block diagram, whereas Fig. 17 shows the principal components of the FKBC.

As shown, the three principal blocks of the FKBC are the fuzzification module, the inference engine and the defuzzification module. The fuzzification module includes a normalization block for mapping the physical values of the control state variables onto the normalized domain. This block also maps the normalized control output variable onto the physical domain. The knowledge base of the fuzzy controller consists of a database and a rule base. The former includes information on the membership functions and the normalization/denormalization scaling factors. The rule base represents the control policy adopted. For implementing fuzzy control on the helicopter, the process state variables are the tracking error e and its gradient \dot{e} .

The control output variable is the control output u . The term sets of e , \dot{e} and u are denoted by $L(e)$, $L(\dot{e})$ and $L(u)$, respectively. They contain the same set of linguistic values, i.e., NL (Negative Large), ZR (Zero) and PL (Positive Large). Thus:

$$L(e) = L(\dot{e}) = L(u) = \{NL, ZR, PL\}. \quad (17)$$

To control the helicopter, users can develop and implement a learning algorithm for the rule base, membership functions, and scaling factors. All these parameters can be tuned online.

After manual tuning, we have a set of appropriate normalization/denormalization scaling factors for the fuzzy variables: 10.0, 50.0 and 1.0 for e , \dot{e} and u , respectively. The interface of the web-based laboratory with fuzzy logic control is shown in Fig. 18.

IMPACTS OF THE WEB-BASED LABORATORY

The web-based laboratory on the helicopter system is currently being utilized in the teaching of a graduate course in the Department of Electrical and Computer Engineering, National University of Singapore. The course is on optimal control systems in which students are introduced to solve both classical optimal control system

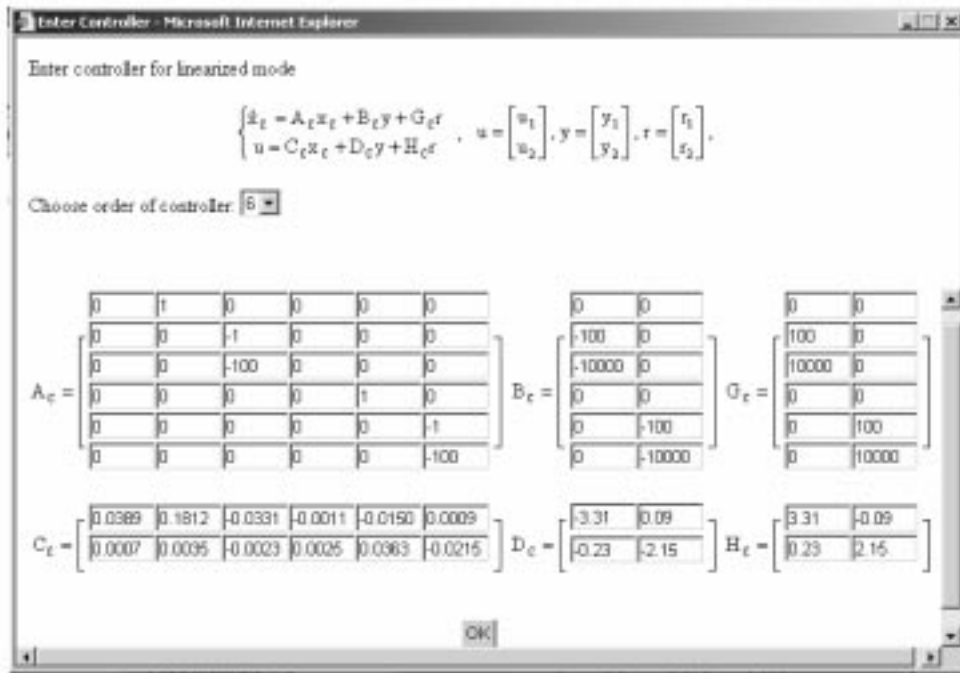


Fig. 15. Interface for user to enter state space controller data.

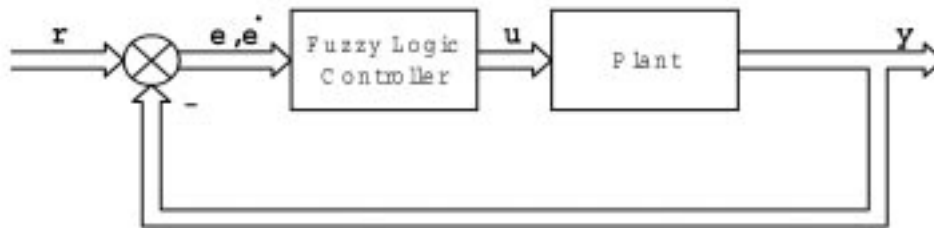


Fig. 16. Block diagram of control system with fuzzy logic controller.

design problems, such as Linear Quadratic Regulator (LQR) control and Linear Quadratic Gaussian (LQG) control, as well as advanced robust control problems, such as H_2 optimal control and H_∞ control. Since half of the class are part-time students with full-time jobs in industries, the course is conducted in evenings once a week with a weekly session of three hours. Due to security reasons and other constraints, laboratories at the University are generally closed after working hours. It was almost impossible in the past to schedule laboratory experiments or to assign homework assignments on hardware implementations. With the help of the web-based laboratory, which is available 24 hours a day, the course instructor is now able to introduce a new teaching element on actual experiments to the class. The web-based laboratory on the helicopter system is used throughout the course homework assignments, in which students are required to solve an actual problem on the control of the helicopter using the above-mentioned techniques learnt in the class, and implement their designs in the actual system through the Internet. The web-based laboratory provides a great deal of flexibility to

part-time students, as they can logon and use it whenever they are free. Feedback and comments from the students are generally very positive. Many students feel that it is superb to see that the controllers they have designed are actually working on the real system.

The web-based laboratory offers an excellent and convenient platform for researchers to test and implement their new algorithms as well. For example, we have currently implemented a control system designed using a newly developed technique. To be more specific, we have employed a so-called robust and perfect tracking technique developed recently in [13] to design a controller, which is theoretically capable of tracking any reference signal with arbitrary fast settling time from any initial condition. However, in general, an infinite gain is needed to guarantee such a performance. With the flexibility of the web-based laboratory, it is simple and easy to tune certain design parameters to meet the control constraints of the physical system with some trade-off in the overall performance. We would like to further emphasize that the web-based laboratory also makes it very easy and meaningful to compare

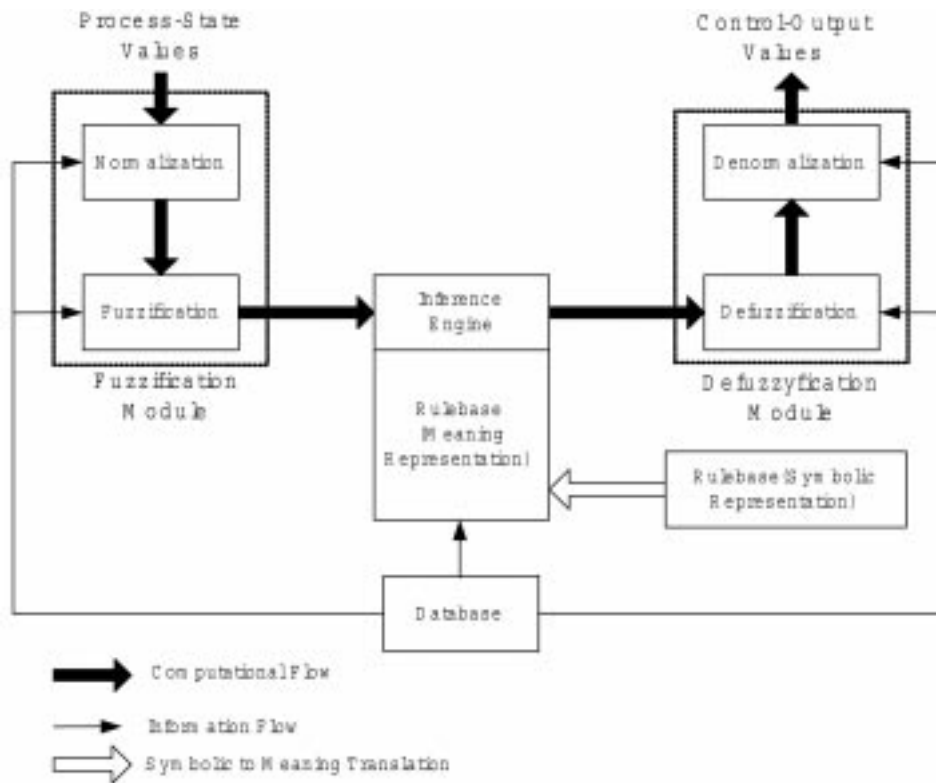


Fig. 17. Principal components of the fuzzy knowledge based controller.



Fig. 18. Interface of fuzzy control in the web-based laboratory.

the performance of different techniques as they are implemented under the same setting.

CONCLUSIONS

We have implemented a web-based laboratory for a two-degrees-of freedom helicopter. Four

different controller interfaces, namely, PID control, decoupled PID control, general state space control and fuzzy logic control, are presented. It is our belief that such a web-based laboratory has a great potential in applications for both teaching and research. Such systems will become more visible in the near future when the bandwidth of the Internet is further increased. The

web-based laboratory on helicopter, together with some other web-based laboratories implemented

earlier, can be accessed online at the website <http://vlab.ee.nus.edu.sg/vlab/>.

REFERENCES

1. S. E. Poindexter and B. S. Heck, Using the web in your courses: What can you do? What should you do? *IEEE Control Systems Magazine*, **19**, 1999, pp. 83–92
2. S. G. Crutchfield and W. J. Rugh, Interactive exercises and demonstrations on the web for basic signals, systems, and controls, *Proc. 36th Conf. Decision and Control*, 1997, San Diego, USA, pp. 3811–3815.
3. A. Simone and J. Tuttle, Designing and implementing college-wide web-based course materials: a case study, *North American Web Conference '97*, New Brunswick, Canada.
4. R. Pascoe, Introducing WWW technology into tertiary teaching: a personal perspective, *North American Web Conference '97*, New Brunswick, Canada (1997).
5. J. Rosenblum and E. Healy, Developing an educational intranet, *North American Web Conference '96*, Brunswick, Canada (1996).
6. C. C. Ko, B. M. Chen, J. P. Chen, Y. Zhuang and K. C. Tan, Development of a web-based laboratory for control experiments on a coupled tank apparatus, *IEEE Trans. Education*, **44**, 2001, pp. 76–86.
7. C. C. Ko, B. M. Chen, S. H. Chen, V. Ramakrishnan, R. Chen, S. Y. Hu and Y. Zhuang, A large scale web-based virtual oscilloscope laboratory experiment, *IEE Engineering Science and Education Journal*, **9**, 2000, pp. 69–76.
8. C. C. Ko, B. M. Chen, S. Y. Hu, V. Ramakrishnan, C. D. Cheng, Y. Zhuang and J. Chen, A web-based virtual laboratory on a frequency modulation experiment, *IEEE Trans. Systems, Man, and Cybernetics*, **31**, 2001, pp. 295–303.
9. M. Shaheen, K. A. Loparo and M. R. Buchner, Remote laboratory experimentation, *Proc. 1998 American Control Conf.*, Philadelphia, USA, pp. 326–329.
10. *User Manual for 2DOF Helicopter*, Quanser Consulting Inc, (2000).
11. P. Eykhoff, *System Identification—Parameter and State Estimation*, John Wiley, New York (1981).
12. W. J. Palm, *Modeling, Analysis, and Control of Dynamic Systems*, John Wiley, New York (1998).
13. B. M. Chen, *Robust and H_∞ Control*, Springer, London (2000).

Chi Chung Ko received the B.Sc. (1st Class Honours) and Ph.D. degrees in Electrical Engineering from Loughborough University of Technology, U.K. He is with the Department of Electrical and Computer Engineering, National University of Singapore. His current research interests include digital signal processing, adaptive arrays, communications and networks. He is a Senior Member of IEEE and has written over 150 technical publications in these areas. He has served as an Associate Editor of the IEEE Transaction on Signal Processing from 1997 to 1999, and is currently serving as an Associate Editor of the IEEE Transactions on Antenna and Propagation.

Ben M. Chen received the BS degree in mathematics from Xiamen University, Xiamen, China, MS degree in electrical engineering from Gonzaga University, Spokane, USA, and Ph.D. degree in electrical and computer engineering from Washington State University, USA, in 1983, 1988, and 1991, respectively. He was an Assistant Professor at State University of New York at Stony Brook. Since 1993, he has been with Department of Electrical and Computer Engineering, National University of Singapore. His current research interests are in systems and control theory, and control applications. He is an author or co-author of seven monographs, and more than 170 technical articles. He was serving on the editorial boards of IEEE Transactions on Automatic Control and Asian Journal of Control. Currently, he is serving as an associate editor of *Control and Intelligent Systems*, a member of international advisory board of *Kuwait Journal of Science and Engineering*, and an associate editor of *Systems and Control Letters*.

Jianping Chen received the B.Eng. degree in automatic control from Shanghai Jiao Tong University, Shanghai, China, in 1993, the M.Eng. degree in control theory and control engineering from Xiamen University, Xiamen, China, in 1999, and the Ph.D. degree in electrical and computer engineering from the National University of Singapore, Singapore, in 2003. His M.Eng. degree was jointly directed at Tsinghua University, Beijing, China. From 1993 to 1996, he was an electrical engineer in South-Cube High-Tech Corporation, Xiamen, China. Since July 2003, he has been with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore, as a research fellow. His current research interests include information security, network security, and control engineering.

Jing Zhang received his M.Eng. degree in electrical engineering from the National University of Singapore, in 2001. He is currently working towards his Ph.D. degree in the Bioengineering Division, the National University of Singapore.

Kay Chen Tan received the B.Eng. degree with first class honors in electronics and electrical engineering and the Ph.D. degree from the University of Glasgow, Glasgow, Scotland, in 1994 and 1997, respectively. He was with the Centre for Systems and Control, and Evolutionary Computing Group, Glasgow, Scotland, before joining the Department of Electrical and Computer Engineering, National University of Singapore, as an Assistant Professor in 1997. He has authored or coauthored more than 100 journal and conference publications and has served as a program committee or organizing member for many international conferences. He currently serves as an associate editor of IEEE Transactions on Evolutionary Computation. His current research interests include computational intelligence, evolutionary computing, intelligent control, and engineering designs optimization.