Redesigning Undergraduate Control Courseware

Timothy Chang[1], Puttiphong Jaroonsiriphan[1], Daphne Chang[2]

[1] *Dept. of Electrical & Computer Engineering, New Jersey Institute of Technology, NJ 07102*

[2] *Counseling Department, Passaic County Community College, N   J 07505*

Keywords: Web experiment, Animation, Simulink, Dynamic System, Control

Abstract: This paper reports on the courseware redesign for a senior undergraduate level control systems analysis course. This course includes analytic methods from both frequency and time domains with emphasis on real world problems. The use of dynamic animation and web experiments to enhance learning and interest constitutes the major changes. A series of lab sessions are introduced to the class to complement the lecture materials and to guide the students into the design project. The use of web experiments and animation provides many advantages: better visual effects, improved communications, and higher interest levels. Student response has been very positive. A number of recommendations are made in this work based on instructor observation and course evaluations.

(I) Introduction

Control system analysis is a multidisciplinary subject encompassing almost all fields of engineering applications. However, the traditional treatment (in the sense of teaching pedagogy) of this subject tends to be highly theoretical and mathematical with heavy emphasis on equation derivation and algorithmic development. Such approach is convenient from the instructor's point of view but may not be beneficial to the students who are classified as sensing types (MBTI), or visual (ILS), or concrete experience Kolb's [11] model.

It is observed that about 70% of learners are not analytic learners. Kolb's experiential learning theory advocates a holistic approach to combine experience, perception, cognition and behavior. This curriculum is designed to adopt Kolb's theory and to adapt to students' learning needs. These class activities allow students with various learning styles to move through stages of experiential learning for one would learn best when one uses all four processes: 1) Concrete Experience, 2) Reflective Observation, 3) Abstract Conceptualization and 4) Active Experimentation.

One of the major functions of education is to shape students' attitude toward learning and to develop effective learning skills. The authors hope to accomplish these objectives by using web experiments, dynamic animation, and team project. It is not merely about including practical experiences but utilizing these experiences to induce higher levels of learning. Furthermore, students will be exposed to the importance of team work, working collaboratively through individual differences, which is an integral part of real work scenarios.

In the past decade, the availability of student dynamic simulation and analysis software provided the first step towards closing the gap between classroom and "real-world" experience. The term animation covers a broad range of software applications such as kinematics and dynamics. For kinematic animation, use of keyframing and motion capture constitute the primary mean of driving

the animation sequence (e.g. Poser, 3D Studio Max, Jack, etc.). Emphasis of kinematic animation falls on "life-like" quality and stunning graphics. It is therefore as much art as science. Dynamic animation is driven by the outputs of a simulation engine to provide a 2D or 3D display of the physical characteristics of the application. The output may contain simplified geometric objects such as line, rectangles, circles, etc. The benefits of dynamic animation include: (1) better interpretation of the results of the simulation so that the students can actually visualize the execution of the system/subsystem, (2) more efficient communication of the results, and most importantly, (3) improving student interest in the subject materials.

A number of researchers have reported their effort in using animation to enhance design and/or education effectiveness. For example [1] described an engineering animation tool that included motion control development. In [2], a real time simulation/animation tool was developed to facilitate the evaluation of active suspension systems. In [3], animation of flexible manufacturing system was carried out in conjunction with modeling and control. Arrival of student versions of graphical simulation software such as Matlab, VisSim, and LabVIEW sparked interests in adding an animation stage to the simulation. Such efforts were described in [4], [5], [6], and [9]. In [8], [10], a Visual C++, Direct-3D based software was generated for interactive modeling, simulation, animation, and control of dynamic systems. In [7], the ubiquitous world wide web was utilized to implement computer animated simulation instruction modules. This paper describes a similar approach to the past effort, i.e. using Matlab/Simulink as the numerical engine. However, the Matlab Animation toolbox, a freely downloadable software that works seamless with Simulink (professional as well as student versions), is used to generate simple dynamic animation for the purpose of enhancing student learning and appreciation of "real-world" dynamic systems. Furthermore, the simulation/animation component is directly integrated into the course so that progressive learning and coordination with the lecture materials can be carried out.

It is well established that hands-on experience can also significantly improve student learning and interest level in the course materials [12]. This is especially important in control analysis courses that are theoretical with a high degree of mathematical analysis. However, a number of constraints prevent the broad base integration of experiments into engineering courses. For examples, laboratory facilities/hours, safety issues, and software development effort are a few of the limiting factors that most instructors would encounter. With the increase in bandwidth through high speed and ISDN data lines, web-based distance experiments can play a significant role in supporting the learning process by providing a 24/7 laboratory with real time experiments so that the students can draw realistic conclusions and gain insight for real-life problem solving.

A number of web-based distance experiments are already in place. Earlier systems [13], [14] rely on custom software development e.g. cgi/Perl, Java/C++, etc. Maintenance and upgrade of the codes become problematic. Furthermore, most earlier system lack real-time streaming capabilities due to the client side control privilege. The more recent implementations of the web experiments are based on existing software platforms such as LabVIEW or Matlab. In [15], [16], and [17], various web experiments are supported by the Wincon/Simulink combination package through TCP/IP protocol. In [18], distance experiments in process control and dynamics are made available via LabVIEW interface. In [20], monitoring of distributed processes are carried out on LabVIEW TCP/IP VI (Virtual Instrumentation) using the HTTP web server.

While these approaches are meritorious and work well for the intended applications, to be effectively applied to an undergraduate environment, distance experimental system should possess the following characteristics:

1. Efficient interactive/multi-user operations.
2. Machine independent (currently developed for the PC platform).
3. Secure operations.
4. Graphics User Interface compatible.
5. High processing bandwidth with true real-time data streaming capabilities.
6. Low cost, user friendly interface.
7. Global resource sharing.

A server/client software based on LabView datasocket technology was developed to for distance experiments. The students download and install a plug-in to a standard web browser such as Microsoft's Internet Explorer and access the experiments on a 24/7 basis. Details of the web experiments are discussed in Section V of this paper.

(II) Course Description
Hands on lab, web experiments, and dynamic animation were introduced into the senior level course "EE482 Instrumentation and Control" in the 2000 academic year and have since been integral parts of this elective course. The old curriculum focused extensively on classical frequency response methods such as complex variables, frequency response methods (Bode, Nyquist, Nichols), stability assessment techniques (Routh-Hurwitz, root locus), performance criteria (sensitivity, steady accuracy, transient response), and compensation (lag, lead). Although it may be argued that the course contents possess educational values, it is generally agreed that significant revision is necessary to reflect the change in technologies and modern engineering career challenges. In particular, computer-aided analysis, state space methods, and nonlinear systems are introduced into the curriculum, replacing lag, lead compensation, Nichols chart, and parts of the performance criteria. That is, the emphasis of the course is on analysis rather than control design which is relegated to a second course "EE486 Control Systems Electives". An outline of the lecture is shown below:
- Introduction to Systems, Review of LaPlace Transform
- Transfer Functions, Signal Flow Graphs, Stability
- Frequency Response of Linear Systems: Bode and Nyquist
- Identification of Dynamic Systems
- Performance Characteristics and Feedback; Case study: positioners, actuators, and sensors
- Properties and Solution of State-Space Systems
- Modeling of Physical/Biological/Nano Processes, Linearization
- Stability Assessment: Routh-Hurwitz
- Root Locus Method
- The Nyquist Stability Criterion
- Describing Functions and Limit Cycles
- Application of Describing Functions
- Review and Project Presentation

The revised contents can potentially be highly mathematical and run into the same problems of being disconnected from the real world. Therefore, a complementary laboratory session is added to introduce limited hands-on experience for the class. The laboratory engine is based on Mathwork's Matlab/Simulink package and consists of the following topics:

- Introduction to Matlab
- Matlab Differential Equation Solvers
- First and Second Order Linear Systems
- Introduction to Simulink
- Common Nonlinear Systems and Simulation
- Simulating Chaotic Systems
- Computer Animation
- Project Development

On the other hand, the web experiments focus on identification of first/second order systems, and control of a simple motor drive. The homework assignments comprise of simpler problems that can be hand calculated so that the students can focus on the concept and mechanics of analysis. More in-depth versions of the problems are assigned in the lab sessions where the students team up and tackle the problems using Matlab and Simulink. The project is introduced at mid-term. It is a realistic, multiple degree-of-freedom type (e.g. two link robot, crane with variable loads) with nonlinear dynamics and were generally difficult to "sense" or "visualize" directly. Dynamic animation provides an extra dimension of learning where the students can develop a deeper sense of understanding of the system characteristics and therefore apply creative solutions to the problem. Finally, a full array of support is available to the students:

- Teaching Assistant
- Peer Facilitator
- Open Lab Hours
- Dedicated Website
- Coordinated Lectures and Lab Sessions

(III) Animation and the Matlab Animation Toolbox

The animation toolbox, Ansim, is freely downloadable from Mathworks. It works with both professional as well as student versions of Matlab/Simulink version 4 and above. Installation of the toolbox is somewhat OS dependent. For the student version (5.3+) running on Windows 98 or 2000 system, the preferred destination of the ansim folder is in the work subdirectory of Matlab as shown in Figure 1 below:

Figure 1: Directory Structure of the Ansim Toolbox

The corresponding Simulink Library Browser window will then correctly reflect the path as shown in Figure 2:
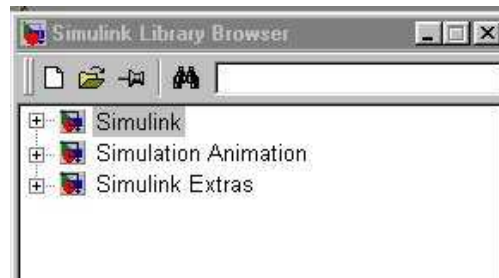

Figure 2: Simulink Library Browser Window Showing the Animation Toolbox

Figure 3: Animblk Block and the Animation Window

Animblk.mdl in the ansim folder, shown in Figure 3, is a self-contained block that handles 2D object animation based on the Cartesian coordinates generated by Simulink. As shown in Figure 3, there are five basic elements in animblk: text, line, rectangle, dot, and patch. The text element enables the user to move, zoom, color, and rotate a text string. The line element has three vertices defined as (x1, x2, x3) and (y1, y2, y3). These parameters are ported into animblk using a mux element in the Simlink Signals and Systems Library. The line width and color can also be specified by the simulation output. Similarly, the patch object is defined by a series of matching x, y coordinates: (x1, x2, …) and (y1, y2, …). The rectangles, on the other hand, are specified by the lower left x, y coordinates, height, and width. Finally, the dot element can vary in position (as specified by the x, y coordinates), size, and color.

A simple example consisting of a second order system (DC motor) under proportional feedback control is shown in Figure 4. In this example, the students can vary the feedback gain "K" and observe the effects on the output angle which exhibits overdamped, critically damped, and underdamped characteristics as "K" is increased in accordance to the root locus diagram. The

conventional output is shown in the scope display. However, the animation screen shows a single rigid link attached to the shaft of the motor so that the link rotates according to the output angle of the motor shaft. The students can now easily visualize the physical characteristics of the motor and draw conclusions that are otherwise less obvious. For example, some students immediately point out that excessive overshoot can be a potential safety problem to both the hardware and people standing near it. It should be noted that the output angle must be converted into the respective Cartesian coordinates for the line element according to (0, L cos ϕ), (0, L sin ϕ) for the x and y vectors. A second example illustrates the motion of a physical pendulum governed by a second order nonlinear differential equation. Both the line element (for the string) and the dot element (for the bob) are used for the animation. Again, the animation output provides a better "feel" for the dynamics of the pendulum (Figure 5).
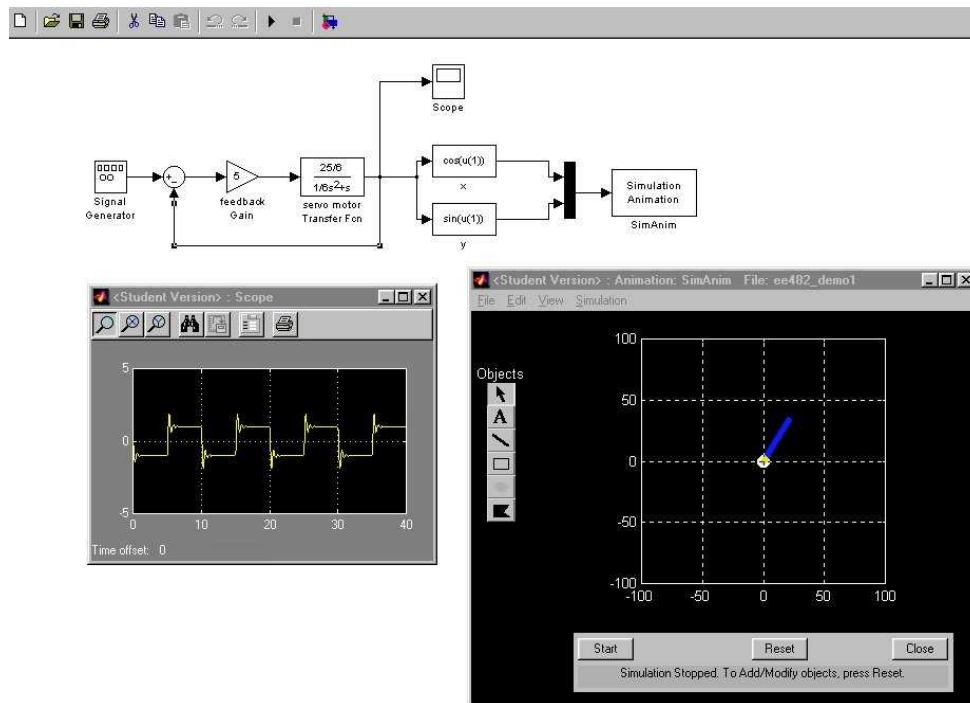

Figure 4: Animation of a Second Order Linear Feedback System
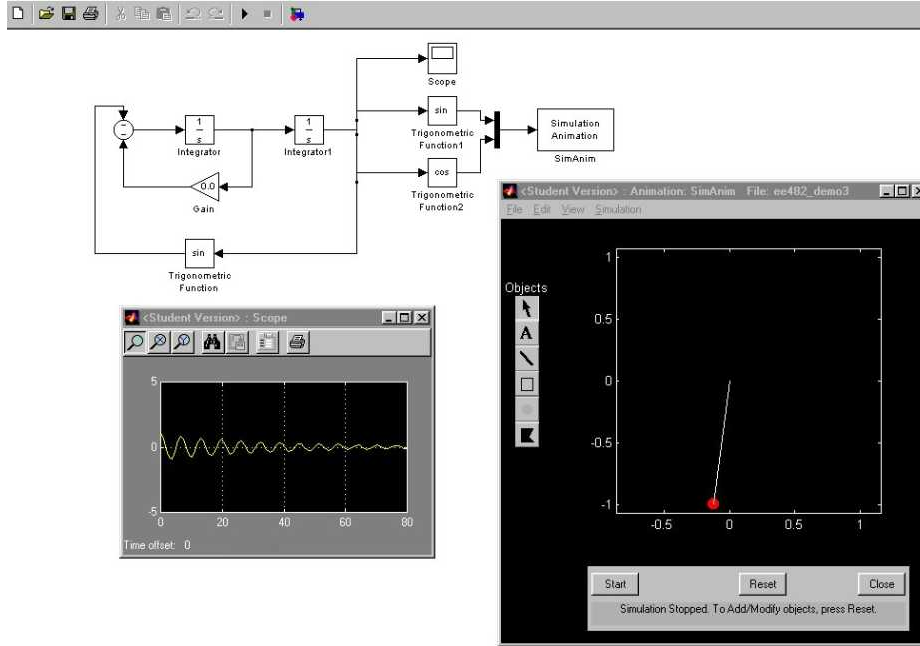
Figure 5: Animation of a Physical Pendulum

(IV) Project Description

The project is an integral part of the course. It accounts for 15% for the total grade and it is assigned at mid-term to the class teams each consisting of 2 students. The project is first broken down into the sub-components for the teams to work on during the lab sessions. That is, simulation block build, debug, and partial animation. The teams are then required to integrate the system together, devise an strategy to coordinate the motion, analyze the results, and produce a parts list and specifications.

Two types of projects have been assigned over the past semesters: a two-link rotary robot and a crane system. Both systems are described by a set of nonlinear differential equations.

Dynamics of the two-link robot are described by the following equations:

$$\tau_1 = m_2 l_2^2(\ddot{\theta}_1 + \ddot{\theta}_2) + m_2 l_1 l_2 c_2(2\ddot{\theta}_1 + \ddot{\theta}_2) + (m_1 + m_2)l_1^2 \ddot{\theta}_1 - m_2 l_1 l_2 s_2 \dot{\theta}_2^2 - 2m_2 l_1 l_2 s_2 \dot{\theta}_1 \dot{\theta}_2$$
$$\tau_2 = m_2 l_1 l_2 c_2 \ddot{\theta}_1 + m_2 l_1 l_2 s_2 \dot{\theta}_1^2 + m_2 l_2^2(\ddot{\theta}_1 + \ddot{\theta}_2)$$
$$c_i = \cos\theta_i, i = 1, 2$$
$$s_i = \sin\theta_i, i = 1, 2$$

where
l_1, l_2 lengths of links 1 and 2 in meters
m_1, m_2 mass of motor 1 and 2 in kg
$\tau_1 \tau_2$ output torque of the motors 1 and 2 in N-m
θ_1, θ_2 angles of links 1 and 2 with respect to the neutral axis in radians

The objective of the two link robot project is to simulate the system and devise a strategy so that the tip of the robot moves on the circumference for 90 degrees as shown in Figure 6 which is captured from the animation:
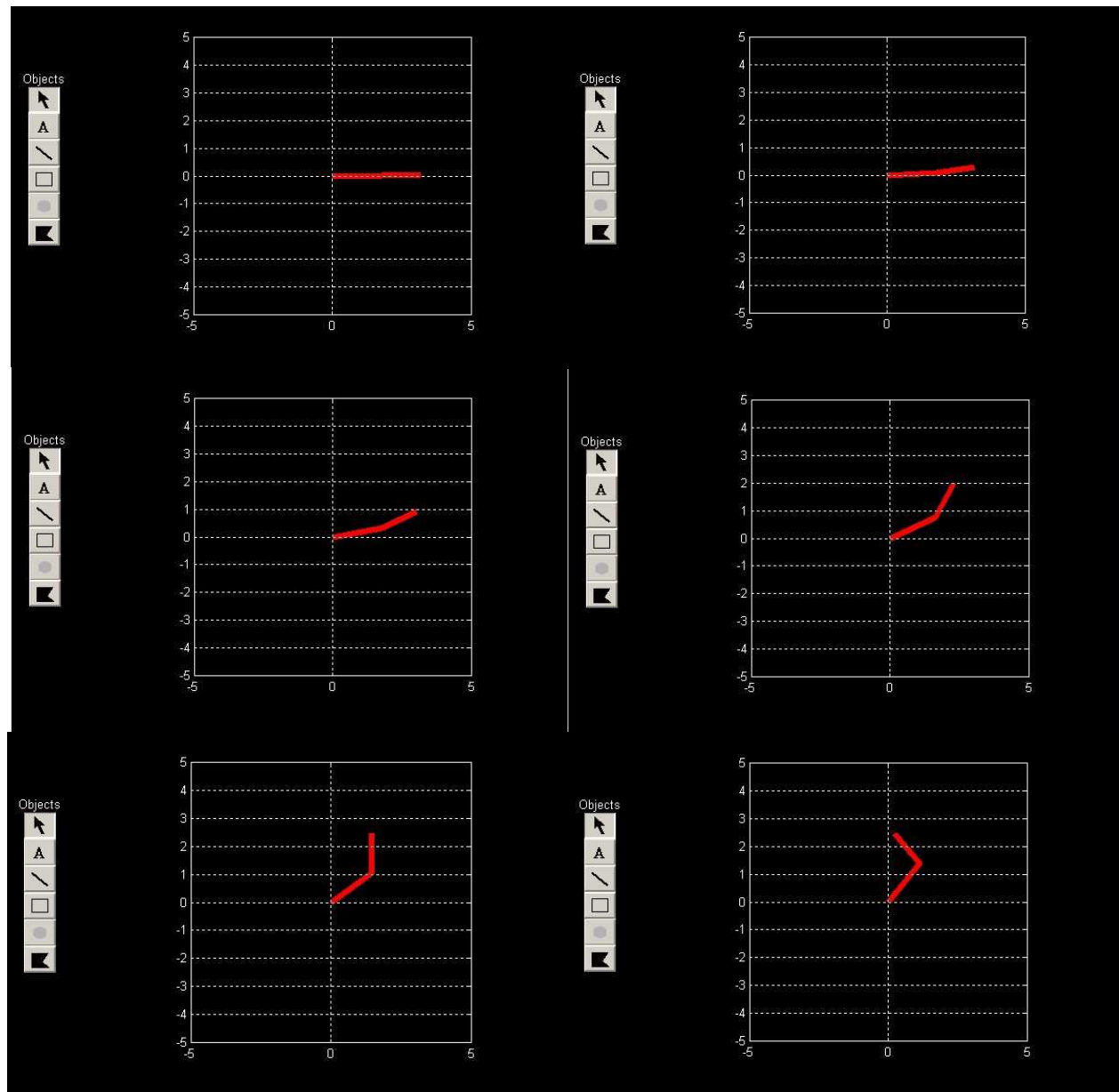
Figure 6: Animated Motion Sequence of the Two Link Robot

A typical Simulink block diagram is shown in Figure 7. Construction of the simulation block diagram requires careful analysis of the equations and coding of the function blocks. This is not a trivial task for undergraduate students that encounter dynamic simulation/animation for the first time. Therefore, three lab sessions were dedicated to the project development. It is especially important that the team members work in a cooperative and proactive manner.
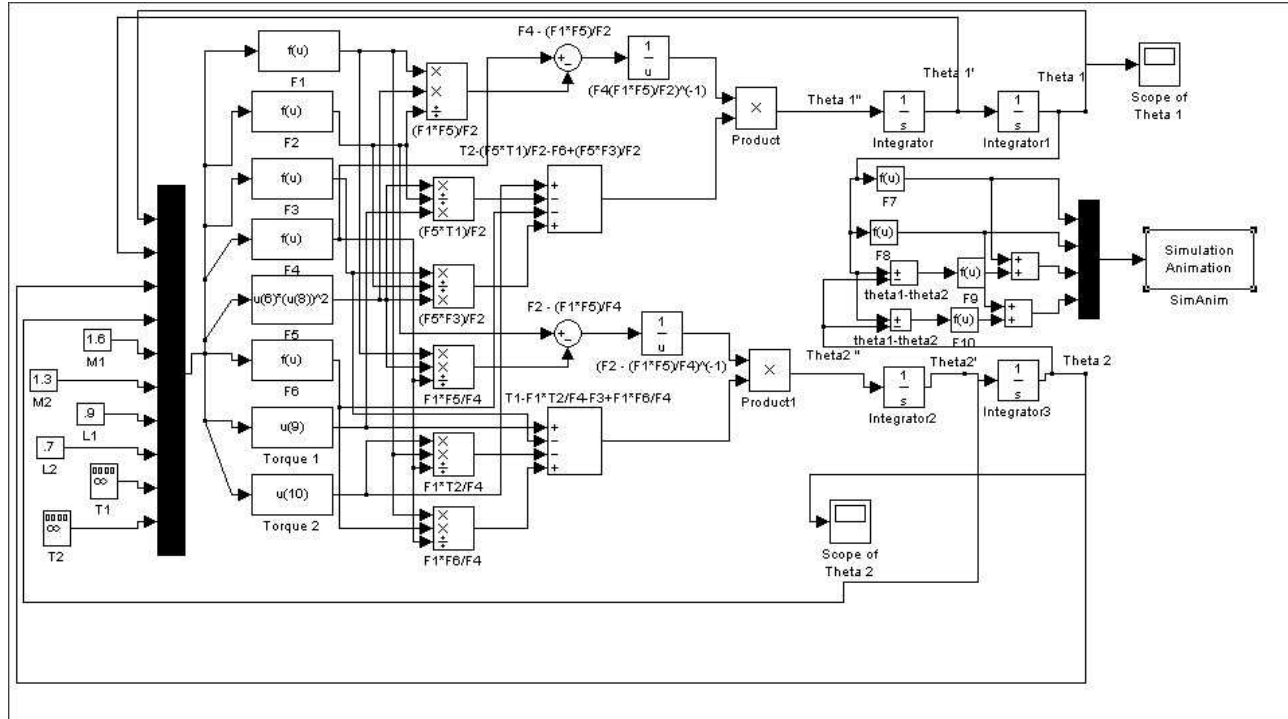
Figure 7: Simulink Block Diagram for the Two Link Robot

Another project (given in the subsequent semester) is a crane system that consists of a crab, a winch, and a cable assembly. The equations of motion are given by:

$$u_1 = (m_L + m_c)\ddot{x}_1 + x_5\ddot{x}_3 m_L \cos x_3 + \ddot{x}_5 m_L \sin x_3 + b\dot{x}_1 + 2\dot{x}_3\dot{x}_5 m_L \cos x_3 - x_5\dot{x}_3^2 m_L \sin x_3$$

$$0 = \ddot{x}_1 x_5 \cos x_3 + x_5^2\ddot{x}_3 + gx_5 \sin x_3 + 2x_5\dot{x}_5\dot{x}_3$$

$$u_2 = m_L\ddot{x}_1 \sin x_3 + m_L\ddot{x}_5 - m_L g \cos x_3 - m_L\dot{x}_3^2 x_5$$

where
m_L	load in kg
m_c	mass of crab in kg
x_1, x_2	crab position, velocity in m and m/s
x_3, x_4	cable angle and angular speed in radians and rad/s
x_5, x_6	cable length and rate of change in m and m/s
u_1, u_2	crab and winch controls
g	$9.8 m/s^2$
b	viscous damping

The objective of this project is to simulate the system and devise a strategy to pick up two loads and release them in the final drop zone as shown in Figure 8 for the motion captured animation sequence. In order to do this, the students must coordinate the motion of the crab and winch while avoid hitting the obstacles.
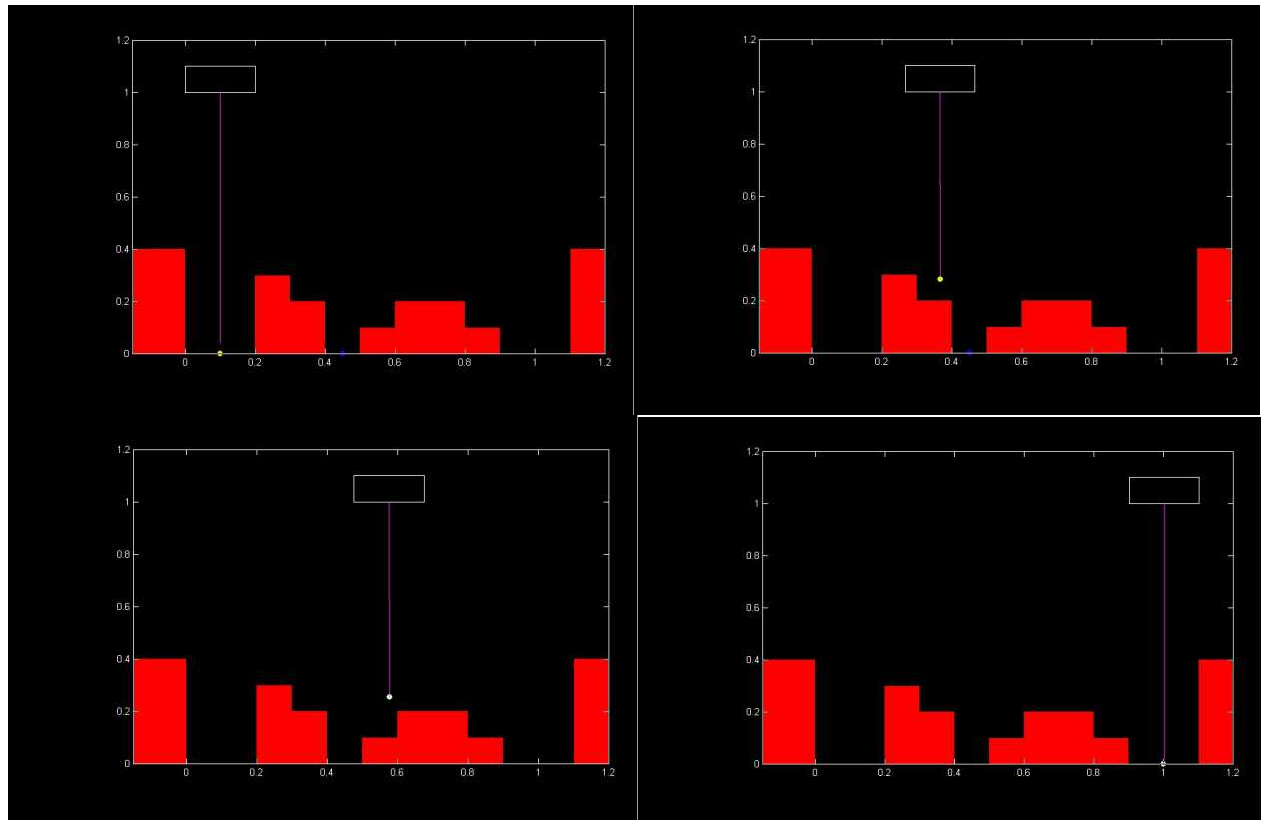
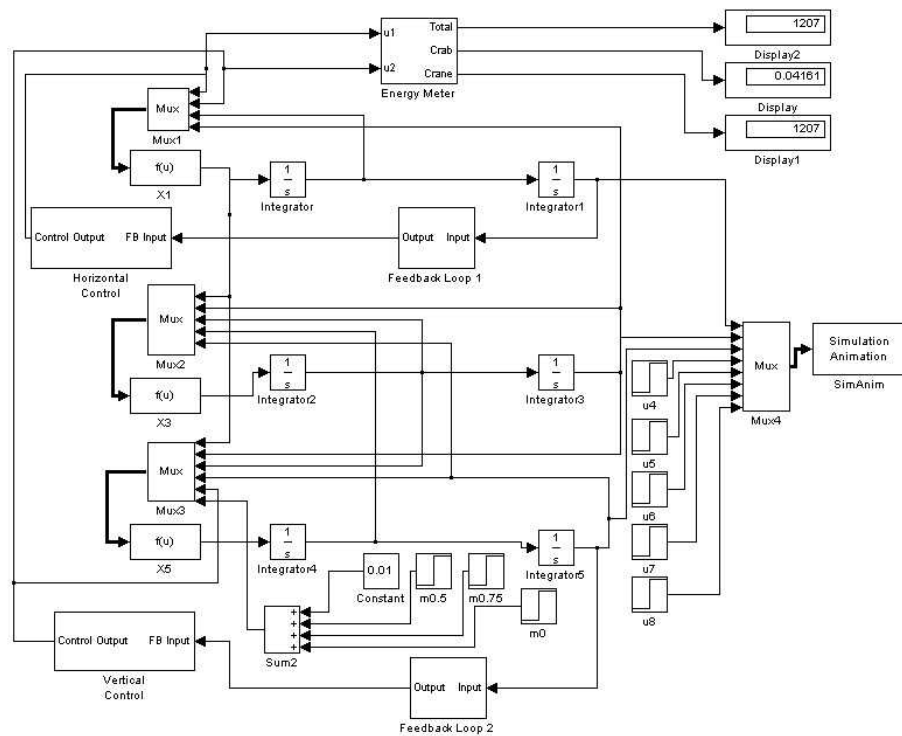**Figure 8:** Crane System Captured Animated Sequence

Figure 9: Simulation Block Diagram for the Crane System

A typical simulation block diagram is shown in Figure 9. It is of interest to compare the animation output with the traditional time trajectory plot given in Figure 10. Although the trajectories yield accurate measurement data, the animation produces a much better description of the dynamics of the system and the project results can also be easily communicated to both technical and non-technical audiences.
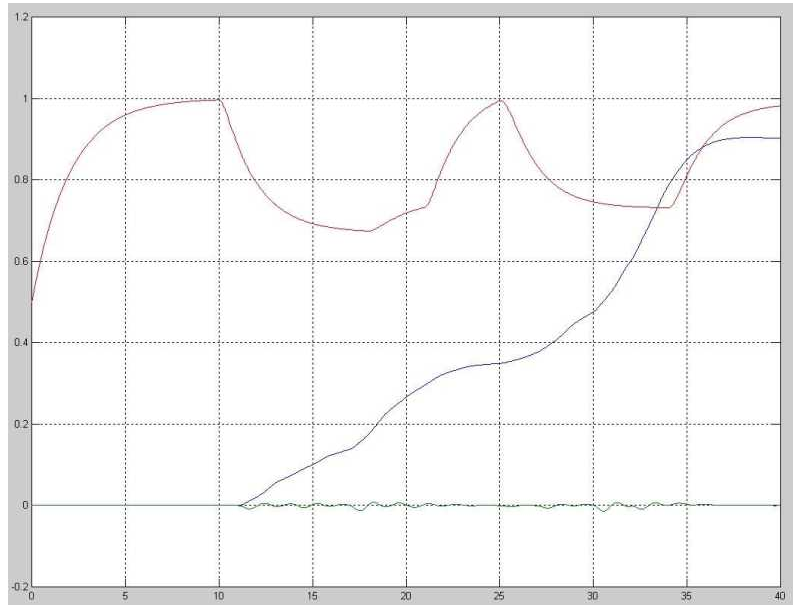

Figure 10: Traditional Trajectory Plots of the Crane System Depicting Variables x1 (middle),x3 (bottom), and x5 (top)

(V) Web Experiments

To participate in the web experiments, the students connect their PCs to the server located in the NJIT Real-time Control Lab by using Web Page Plug-In and applying IP address of the server. The students (client) connect to the server containing DataSocket reader and writer components that automatically control and display the real time data. This way, the students can control and monitor experiment in real-time and enjoy the full benefits of telepresence. In terms of connectivity requirements, the students need to have Internet access either through LAN or WAN. For off-campus access (e.g. dial-up or DSL) the students must have a valid NJIT account and a VPN (Virtual Private Network) program to get across the firewall. Detailed descriptions of the server/client designs are given in [19], [21]. A description of the web experiments is now given:

1. **First and Second Order system identification:** This experiment deals with the characteristics of first and second order systems. As such, the experiment is a simulation to focus on the linear dynamics and to get the students ready for the second web experiment on DC motor control. A simulation loop is set up on the server and the students are to identify the open loop transfer functions (first and second order) by means of the input-output data. The system block diagram is shown in Figure 11. The client (student) and server screen captures are shown in Figures 12 and 13, respectively. After choosing the first or the second order system, the students then specify the type of inputs to the system as well as the loop gain in order to establish the closed loop characteristics of the system. When students are

satisfied with the results, they can save the data by selecting the files to save in the local drive. The results are then analyzed to derive the open loop transfer functions. This experiment is introduced around the fifth week of the course, after coverage of the topic "Identification of dynamic systems".
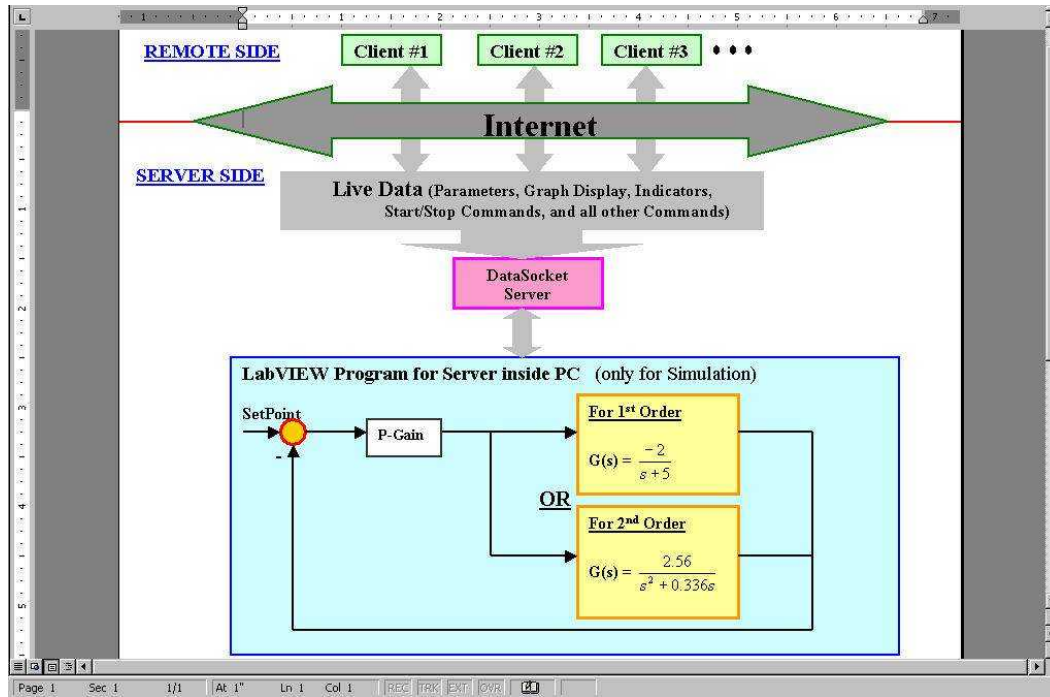

Figure 11: Web experiment1: First and second order systems identification
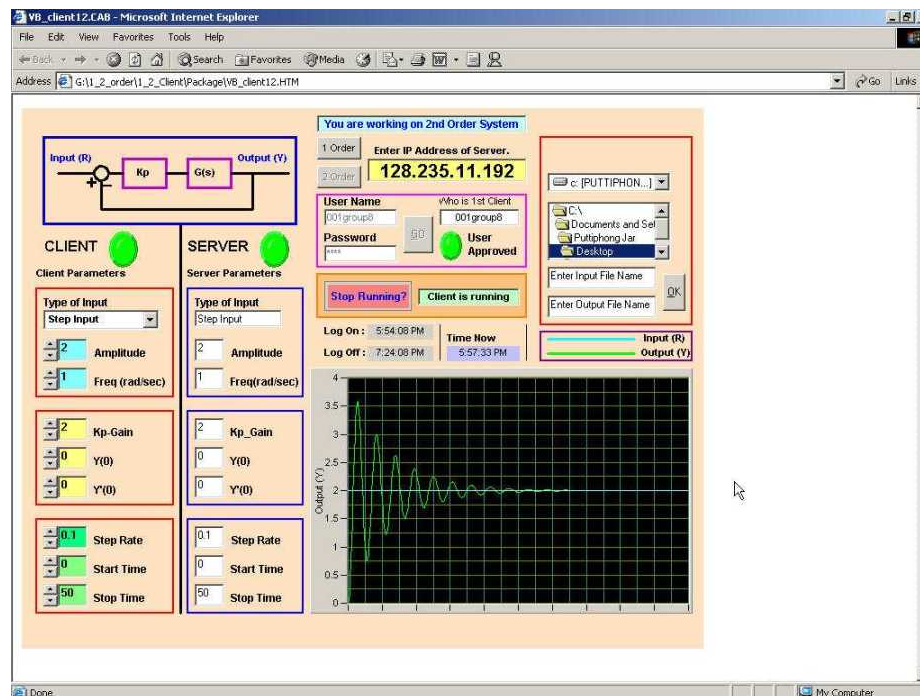

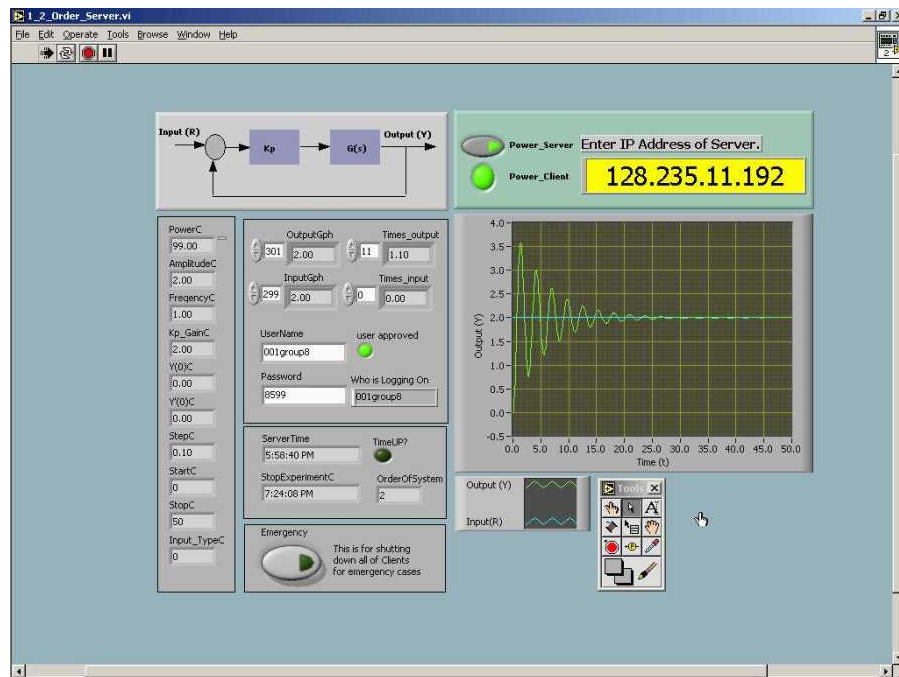**Figure 12:** Web experiment 1: Client screen capture

Figure 13: Web experiment 1: Server screen capture

2. **Position Control for DC Motor.** This experiment demonstrates motor control and the effects of the non-linearity present in a mechanical system. The set up consist of a standard DC motor/potentiometer/amplifier combination as shown in Figure 14. Unlike the ideal linear systems of experiment 1, the responses of non-linear system under proportional control does possess non-zero steady state error and exhibits limit cycles resulting from the dead zone and parasitic dynamics. The dead zone represents frictions in the motor bearings and has been broadened to increase the observable effects. The students are also provided with the corresponding simulation block diagram as shown in Figure 15. The goals of this experiment are (1) to determine the motor parameters and dead zone characteristics through a combination of analysis and simulation and (2) compare the response of the nonlinear system with a linear plant. Screen capture of the client program is shown in Figure 16. This experiment is introduced around the tenth week of the course, after topics such as performance criteria and nonlinearities are introduced. It should be noted again that the primary goal of this experiment is to introduce the students to the nonlinear characteristics and the process of identifying the nonlinear plant. Control of the motor through more advanced control will be discussed in a follow-up course EE486 Control Systems Electives.
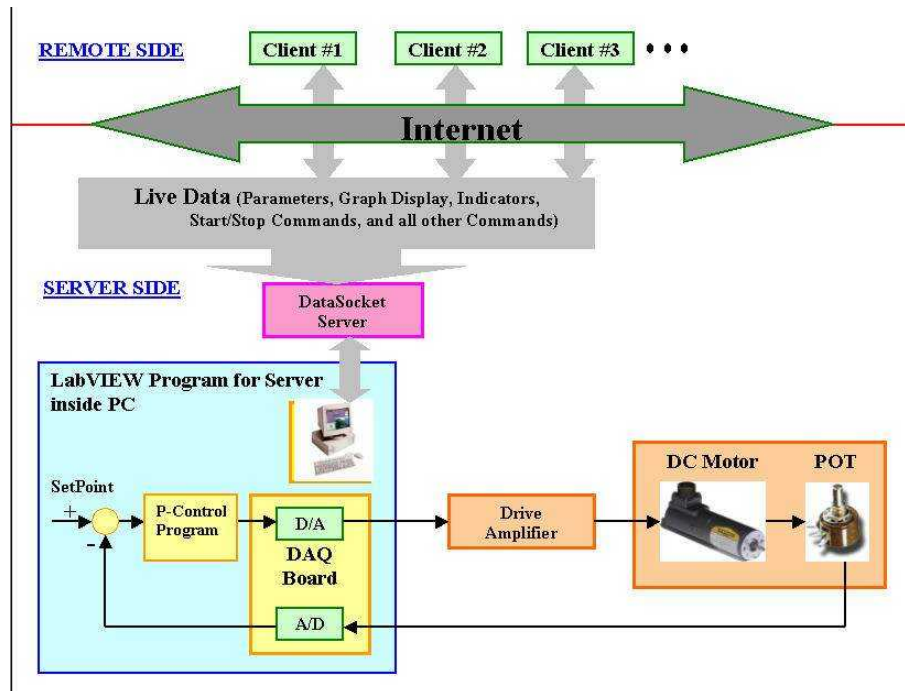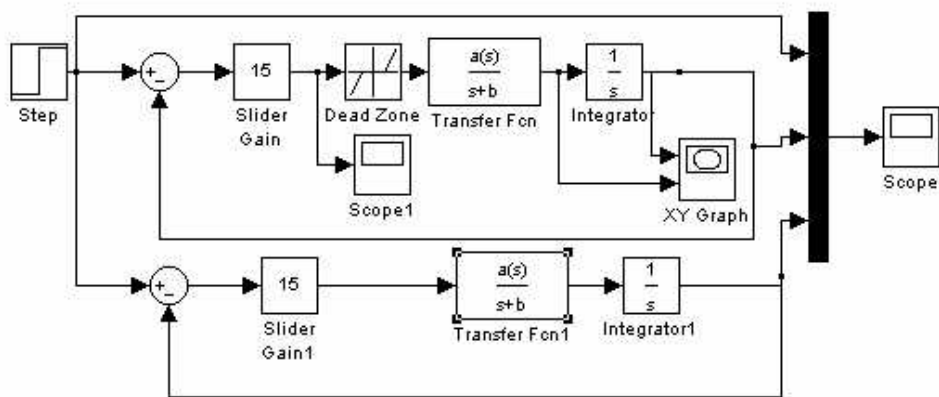
Figure 14: Web experiment 2: DC motor control

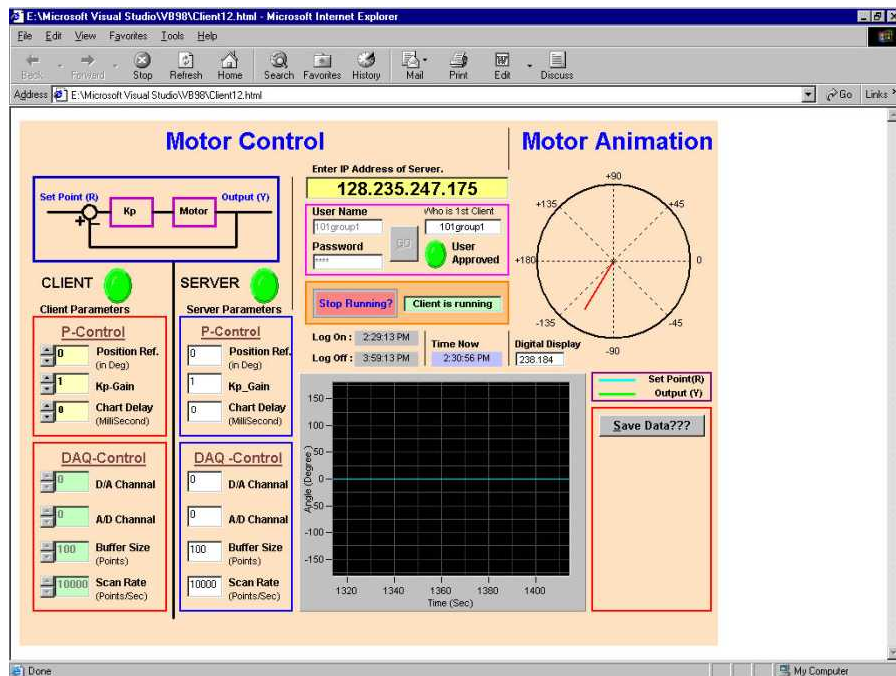**Figure 15:** Simulation block diagram for the DC motor experiment

Figure 16: Web experiment 2: Client screen capture

For Experiment 1, the concept is relatively simple and there are relatively few surprises. Most students are able to come up with the open loop transfer functions using either step response (most popular) or frequency response methods. The identification experiment serves as a summary of linear systems characteristics. However, unlike the traditional approach of synthesizing the output according to the plant model, identification requires a higher level of understanding of system properties and the effects of feedback on input-output relationship.

For Experiment 2, the DC motor control system, the situation is different. Most students have difficulties initially in dealing with the system because of a number of factors: (1) the experiment is not 100% time-invariant/repeatable, (2) the response curves exhibit severe nonlinear behavior such as limit cycle and "uncertain" steady state values. Nevertheless, the lecture materials and simulation block diagram were helpful in helping the students to "sort out" the nonlinear dynamics.

Overall, the student experience has been very positive and the stated goals of strengthening learning and enhancing interest level have been achieved. However, further development is necessary to fully optimize the experiments.

(VI) Student Experience
Having piloted this new course format for four semesters, student experience can be described as "apprehensive at the beginning" and "exhilarative at the end". Although many students are initially concerned with the difficulties and workload of the revised course, this concern gradually subsides and a marked increase in interest level is observed. From the course evaluation, students are particularly positive about the following:

1. Introduction of new software technology such as the animation toolbox.

2. Relevance of course contents to "real-life" experience.
3. Challenges and rewards of the project.

Table 1 below summarizes the major statistics of the revised course since 2000.

Table 1: Course enrolment pattern and student response

Semester	Number of students	"Overall education value of the course" (out of 4)
2000 spring	26	3.57
2000 fall	20	3.38
2001 spring	19	3.57
2001 fall	37	3.73
2002 spring	22	3.29
2002 fall	30	3.75
2003 spring	22	3.67
2003 fall	28	NA

Another indicator about the course is that enrolment for the follow up control electives
EE484/EE496 (lecture/lab) is significantly up since 2000. Prior to that, the elective pair could
barely run once a year. Now the pair can run every semester with good enrolment. Finally, although
the statistical base is still too limited to draw any definitive conclusions at this point, it is
nevertheless observed that:
1. Success rate of the project does correlate strongly with the rest of the lecture grades and work
   pattern. Students submitting their class works on a timely basis and scoring above average
   grades in the tests have the least difficulties in completing the project. Therefore, a highly
   structure project schedule is recommended.
2. Most students prefer feedback approach than open loop strategies. This is because the plants
   (robot or crane) are marginally stable and adding a bit of proportional feedback does simplify
   the design. The use of feedback in this project, however, will cause some loss of physical
   insight into the problem and some open loop strategies should be mandatory.
3. General computer skills (e.g. web surfing, games, etc.) do not necessarily translate into project
   success. This is because the project is more than just programming: knowledge of analysis and
   characteristics of the plant are equally important. Coordination of lecture and lab sessions is
   vital.
4. Team spirit is critical. There are quite a few cases of break down in cooperation between
   members. Nevertheless, the team approach has its merits and will be continued. It may be
   necessary to apply additional partnership criteria e.g. Myers-Briggs Type Indicator to improve
   teaming arrangements and communication among members.

(VII) Conclusions
In this paper, courseware redesign for a senior level course on control system analysis is reported.
Besdies revising the course contents, a series of lab sessions are introduced to complement the
lecture series with the review of Matlab, introduction of Simulink, Ansim, web experiment, and
culminating in the project design. The project is a real-life system with high degree of freedoms
and nonlinearities. The use of animation added an extra dimension of visual communication that
brought the students closer to the physics (rather than just the mathematics) of the problem. Student

response has been very positive despite the extra work and, at times, the difficulties in maintaining coherent team spirit. Interest level in the course has also been increased, as evident in the course evaluations. The Kolb's experiential learning cycle encourages students to observe, think, analyze, synthesize, and apply what they have learnt. The authors are pleased to report the course revision has moved toward the direction of shaping students' learning attitude and developing effective learning skills.

References

[1] Neelamkavil, F., O'Tuathail, E., "System for engineering animation with automatic motion control," Computer-Aided Engineering Journal, v 8, n 3, Jun, 1991, pp. 103-115.

[2] Huang, N., Cheok, K. C., Horner, T. G., Settle, T., "Real-time simulation and animation of suspension control system using TI TMS320C30 digital signal processor," Simulation, v 61, n 6, Dec, 1993, pp. 405-416.

[3] Dimirovski, G.M., Radojicic, P.C., Markovic, N.B., Llliev, O., Gough, N. E., Zakeri, A., Henry, R.M., "Modelling, control and animated simulation of complex processes in robotised FMS," Proceedings of the 20th International Conference on Industrial Electronics, Control and Instrumentation. Part 2 (of 3), Sep 5-9 1994, Bologna, Italy, pp. 1141-1146.

[4] Watkins, John, Piper, George, Wedeward, Kevin, Mitchell, E. Eugene, "Computer animation: A visualization tool for dynamic system simulations," Proceedings of the 1997 ASEE Annual Conference, Jun 15-18, 1997, Milwaukee, WI, USA.

[5] Brooking, Cole J, and Smith, Donald A., "Simulation and animation of kinematic and dynamic machinery systems with MATLAB," Proceedings of the 1998 Annual ASEE Conference, Jun 28-Jul 1, 1998, Seattle, WA, USA.

[6] Tilbury, D., Messner, W., "Controls education on the WWW: Using MATLAB for control design, simulation and visualization," Proceedings of the 1998 ASME International Mechanical Engineering Congress and Exposition, Nov 15-20 1998, Anaheim, CA, USA, pp. 65-70.

[7] Villareal, S., Wynn, C., Eastwood, D., Zoghi, B., "Design, development and evolution of Web-based materials featuring computer-animated simulations,"
Proceedings of the 1998 28th Annual Frontiers in Education Conference, FIE. Part 2 (of 3), Nov 4-7 1998, Tempe, AZ, USA, pp. 588-593.

[8] Lim, C. I, Metzger, R.P. Jr., Rodriguez, A. A., "Interactive Modeling, Simulation, Animation and Real-Time Control (MoSART) twin lift helicopter system environment," Proceedings of the 1999 American Control Conference, Jun 2-Jun 4 1999, San Diego, CA, USA, pp. 2747-2751.

[9] Ravn, O., Larsen, T., Andersen, N., "Simulation and animation in Simulink and VRML," Proceedings of the 1999 IEEE International Symposium on Computer Aided Control System Design, Aug 22-Aug 27 1999, Kohala Coat-Island, HI, USA, pp. 120-125.

[10] Rios, C., Lim, C.I., Metzger, R. P., Rodriguez, A. A., "Multivariable analysis and control of a cart-pendulum-seesaw system using an animation tool," Proceedings of the IEEE Conference on Decision and Control, v1, Dec 7-Dec 10 1999, Phoenix, AZ, USA, pp. 825-830.

[11] Kolb, D.A., Experiential Learning: Experience as the Source of Learning and Development, Simon & Schuster Trade, October 1983.

[12] Chang, T.N., Chang, D., ``Graduate Engineering Student Performance Assessment: How learning pattern affects test scores," Proceedings of the ASEE2000, St. Louis, Mo, June 18-21, 2000.

[13] Aktan, B., Bohus, C.A., Crowl, L.A., Shor, M.H, " Distance learning applied to control engineering laboratories," IEEE Transactions on Education, Vol. 39, pp 320 – 326, Aug. 1996.

[14] Shen, H. Xu, Z., Kristiansen, V., Strom, O, and Shur, M, "Conducting Laboratory Experiments over the Internet," IEEE Transaction on Education, Vol42, pp. 180-185, August, 1999.

[15] Exel, M., Gentil, S., Michau, F., Rey, D., "Simulation workshop and remote laboratory: two web-based training approaches for control," Proceedings of the American Control Conference, v5, 2000, 2000 American Control Conference, Jun 28-Jun 30 2000, Chicago, IL, USA, p 3468-3472.

[16] Junge, T.F., Schmid, C., "Web-based remote experimentation using a laboratory-scale optical tracker," Proceedings of the American Control Conference, v4, 2000, 2000 American Control Conference, Jun 28-Jun 30, 2000, Chicago, IL, USA, p 2951-2954

[17] Schmid, C., Ali, A, "Web-based system for control engineering education," Proceedings of the American Control Conference, v5, 2000, 2000 American Control Conference, Jun 28-Jun 30 2000, Chicago, IL, USA, p 3463-3467.

[18] Henry, J., "Running Laboratory Experiments via the World Wide Web," Session 3513 at the 1998 National Meeting of ASEE, Seattle, Washington.

[19] Chang T.N and Hung, D.C. ``Web-based Distance Experiments: Design and Implementation," Proceedings of the International Conference on Engineering Education, Taiwan, August 14-18, 2000.

[20] Tan, K.K., Lee, T.H., Soh, C.Y., "Remotely operated experiment for mechatronics: Monitoring of DCS on the internet," IEEE/ASME International Conference on Advanced Intelligent Mechatronics, AIM, v 2, 2001, 2001 IEEE/ASME International Conference on Advanced Intelligent Mechatronics Proceedings, Jul 8-12 2001, Como, p 1106-1111.

[21] Chang, T.N., Jaroonsiriphan, P., Sun, X. "Integrating nanotechnology into undergraduate experience: a web-based approach," to appear at the International Journal of Engineering Education, Special Issue on Engineering Education in Nanotechnology.

[22] Edwards, H., ``Building an Interactive Web Page with DataSocket,'' Application Note 127, June 1999 Edition, Part Number 341572B-01, National Instruments.

[23] Edwards, H., ``Connecting Measurement Studio User Interface ActiveX Controls to Remote Data,'' Application Note 151, November 2000
Edition, Part Number 341972C-01, National Instruments.