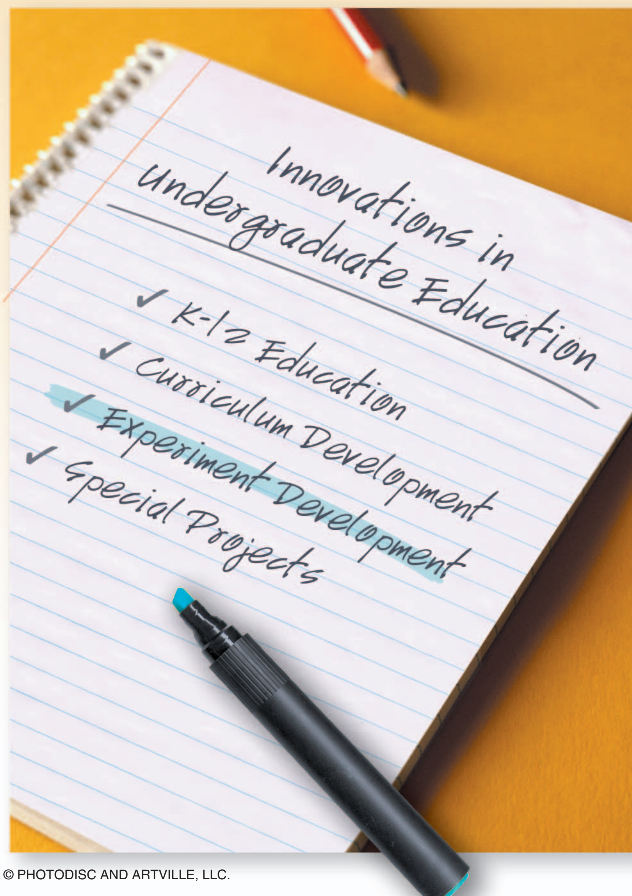
FEATURE

A Laptop Servo for Control Education

A portable process for investigating control system performance and evaluation

By Jacob Apkarian and Karl J. Åström



A control engineer should master theory and have a good understanding of practical control problems. The skill base includes modeling, control design, simulation, implementation, commissioning, tuning, and operating a control system [1], [2]. These skills are becoming more important as control becomes more ubiquitous [3]. While many tasks can be learned from books and computer simulations, laboratory experiments are necessary to develop the full range of skills.

The typical setup for control experiments consists of a physical process with sensors, actuators, power supply, a PC equipped with interfaces, and sometimes a digital signal processor (DSP) board. Control is performed using the DSP or the PC, and the controller is either hand coded or designed using commercially available design tools such as Simulink, System Build, or Labview. Once the design is performed, real-time code is generated and executed on the PC using high-performance real-time software such as WinCon, xPC target, or LabviewRT.

A first-year control course does not usually focus on practical issues, but rather typically focuses on theoretical aspects. Although special lab courses are offered as a complement to the theoretically oriented courses, many students do not take such courses. This reluctance is unfortunate because good experiments can also be a strong motivation to pursue a career in controls.

An introductory course in control with integrated labs has been given at Lund University for many years. This combination of class and labwork has many advantages [4], [5], but even with that approach there is a difference between lectures and labs. Although a student can pick up a book or do a computer simulation at any time or place, experiments are generally restricted in time and space.

0272-1708/04/$20.00©2004IEEE

In this article we describe a portable process that can be used with a laptop computer to investigate control system performance and evaluation. The system can be signed out by the student and taken home, to the library, or café, thus eliminating the need for lab space. The system makes it possible to integrate the theory and practice of control. The experiments can be done concurrently with studies of theory and computer simulation and thus are suitable for practicing engineers who would like to update their control knowledge. The system is completely self-contained, and the process consists of a dc motor and a Quanser integrated controller (QIC). The QIC contains a PIC microcontroller and the peripherals required to interface to the experimental hardware. The process can be controlled using a laptop with no other software than that supplied with the system.

The experiments are designed to maximize system use and expose the user to key industrial and theoretical control issues. A graphical user interface allows the user to download precompiled controllers and plot and tune parameters on the fly. The system also introduces haptics and virtual reality, which provide a "coolness" factor that we hope will arouse curiosity and stimulate students to pursue a career in controls.

The Quanser Engineering Trainer System

The Quanser Engineering Training (QET) system shown in Figure 1 [6] consists of a motor instrumented with an encoder. The motor is driven by a linear power amplifier, and power to the system is delivered from a wall transformer. Signals to and from the system are available on a header as well as standard connectors for control by means of a hardware-in-the-loop (HIL) board. The system can also be controlled by an external PC equipped with a HIL board or by analog controllers implemented on the breadboard. A socket that accommodates a PIC microcontroller is also available. The PIC microcontroller reads the encoder, applies voltages to the motor amplifier, and communicates with a laptop using a serial cable. For the applications described below, the motor is controlled using the PIC microcontroller. A software package (QICii) running on the laptop allows the user to download pre-compiled code and tune and plot parameters on the fly.

Curriculum

A detailed curriculum that demonstrates the relevant characteristics of each control topic was developed to guide students and teachers. Each lab has a prelab preparation section in which the student performs all of the theoretical developments required for the session and performs calculations for parameters that are subsequently used during the experiments. This approach ensures that each student is ready to carry out the experiments.

This system can be signed out by the student and taken home, to the library, or café, thus eliminating the need for lab space.

The prelab activity is followed by an in-lab activity where students perform the experiments in a lab or wherever is convenient. To perform the experiment the student launches the QICii application. A screen capture of a typical QICii session is shown in Figure 2. The manuals direct the student to perform specific experiments using the interactive software. Data is collected by the student for specific activities and entered into pre-formatted tables. These tables facilitate the comparison of results obtained from theoretical derivations and actual performance. The student is then asked to discuss the results.

Experiments

Many different experiments can be performed with the system. In fact, we were surprised to see how many control concepts can be demonstrated using a simple servomotor.

Modeling

Although practicing industrial control engineers do not typically derive models of the system they are controlling,

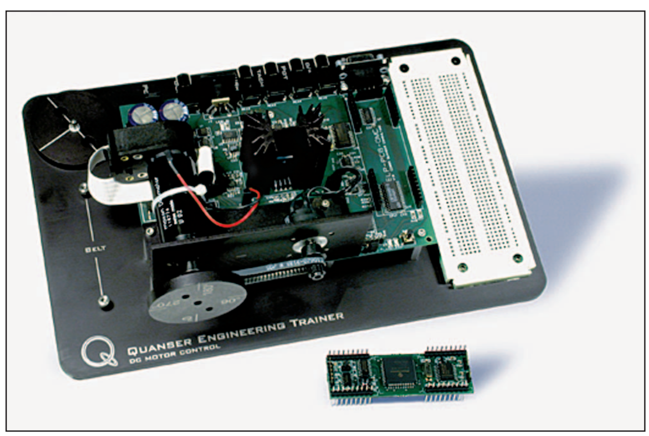


Figure 1. *QET DC Motor Control Trainer. This system can be used with analog controller, HIL/PC-based control, DSP control, and embedded control. This article describes the use of the QET as a portable embedded control system, which performs control experiments using a laptop to communicate with the PIC microcontroller. The PIC microcontroller module (QIC) plugs into a socket on the QET board.*

this experiment stresses the importance of knowing the system before you control it. The students derive a theoretical model of the open-loop system, and parameters of the model are estimated from static data and step responses. A first-order model is simulated in real time, and the response is displayed in parallel with the output of the system. The system is designed in such a way that an excellent model can be derived from first principles. All of the physical parameters can be determined by simple experiments. Real-time open-loop tests are performed, and system parameters are estimated using static and dynamic measurements. A first-order simulation of the derived model is run in real time in parallel with the actual system, and a step response is performed to assess the validity of the identified model.

The experiments can be done with studies of theory and computer simulation and are suitable for engineers who want to update their control knowledge

Speed Control

Motor speed control is a good way to learn PI control. The PI controller is perhaps the most commonly used controller, and students and practitioners of control should be familiar with it. Students are asked to investigate the qualitative properties of proportional and integral action to develop an intuitive feel for PI control. Controllers are tuned by empirical methods of the Ziegler-Nichols type as well as by design to given specifications. Students analyze and test the effect of setpoint weighting, an approach that is often ignored in educational settings but relied on heavily in industrial control. The effect of integrator windup is examined, and an integrator antiwindup scheme is tuned and evaluated. Antiwindup demonstrates that performance can be improved by introducing nonlinearities. Disturbance effects, simulated by direct manual interaction or by a user switch activated by the QIC, are examined, and steady-state errors with triangular reference inputs are assessed. Square wave, sinusoidal, and triangular reference signals can also be discussed.

Robustness

Robustness to modeling errors is an essential property of a good control system. Following the speed control experiment, the student is introduced to sensitivity analysis and stability margins. Sensitivity and complementary sensitivity functions for the speed control system are derived, and the student is guided in designing a controller that is more robust than the PI controller. Sampling delays and filtering effects are taken into account, and the stability gain and phase margins are derived. The margins are then measured using the actual system. The QICii system allows the user to introduce sample delays in the loop and vary the loop gain. Using these features, the system can be driven to instability, and the actual phase and gain margins can be obtained and compared with the theoretically derived values. Disturbance response is also assessed in light of the robustness concepts.

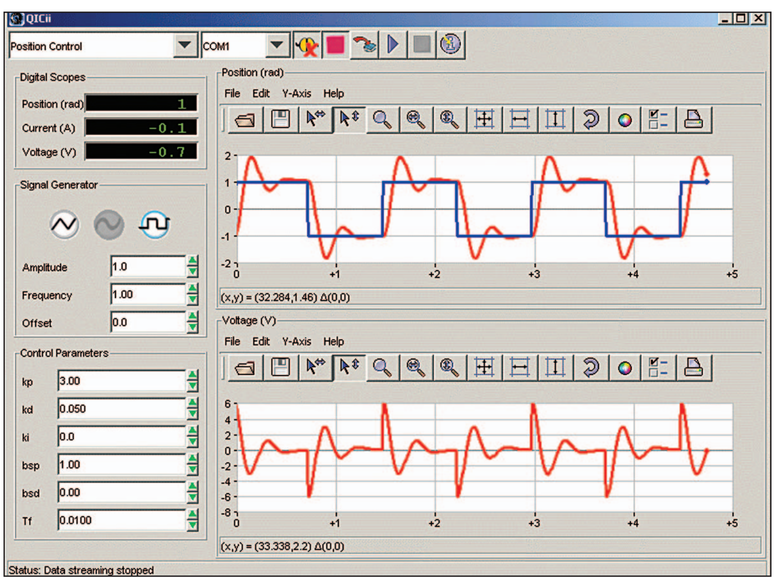


Figure 2. *QICii software. QICii facilitates the download of pre-compiled code to the PIC microcontroller, which performs the real-time control. QICii communicates with the PIC in real time, allowing parameter tuning and data collection. In this example, the system is running a PID position controller.*

Position Control

Control of motor position is a natural way to introduce the benefits of derivative action. The student is asked to design a proportional-integral-derivative (PID) controller to specifications and analyze its response to step inputs, triangular inputs, and disturbances. The controller is implemented in the QIC module, and the user assesses the effects of the three gains on system performance. With derivative action the effect of measurement noise is also clearly visible, which motivates noise filtering. Disturbance response is evaluated with and without integral control. Response to triangular inputs is also assessed.

Haptic Interaction

We have also included an elementary haptics experiment to introduce impedance control using a feedback system. The joint stiffness and damping are derived using a position PD controller. We then simulate a haptic knob by combining a PID position controller with a finite state machine. The student can define detents and step sizes on the motor shaft, which then behaves like a notched knob as a result of software running on the QIC.

The effectiveness of haptics in manual control is also illustrated by a second haptics experiment. The motor shaft is used as an input device to control a virtual ball and beam setup animated on the laptop PC in real time as shown in Figure 3. The motor shaft is used to command the beam angle, and force feedback is used to input different signals to the shaft. The user can select to feel a variety of effects such as ball speed, ball position, and beam texture (ridges on the beam) using the motor shaft. The haptic effect is achieved by applying a voltage to the motor armature proportional to the parameter that the user desires to feel.

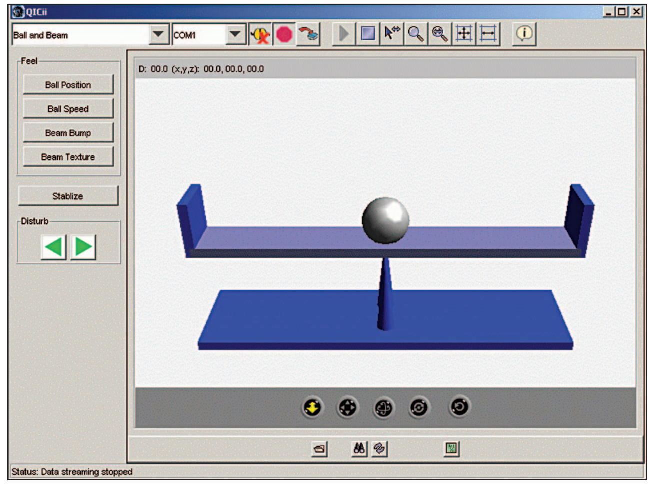


Figure 3. *Screen capture of the haptic ball and beam system. The QET motor shaft is used as an input device to control the angle of the beam. Ball dynamics are simulated on the laptop in real time. The user can select to feel a variety of effects from the simulation by means of the motor shaft. The graphics representation is in 3D and runs in real time.*

Future Developments

During the development of the system and the labs we were surprised by how many control concepts can be illustrated with a servomotor. We have not yet exhausted all of the possibilities. For example, the system can be used to illustrate optimal control to obtain large and fast position changes, while demonstrating the differences between linear and nonlinear controllers. State feedback and observers can also be demonstrated with the QET system.

Conclusion

We have presented a novel portable laptop control experiment, which does not require the space and expense typically required for undergraduate control laboratories. Interactive experiments are designed to cover all of the topics required for an introductory course in controls, as well as a haptics experiment. We believe the system is useful for practicing engineers who wish to update their control knowledge and skills.

References

[1] E.H. Bristol, "An industrial point of view on control teaching and theory," *IEEE Contr. Syst. Mag.*, vol. 6, no. 1, pp. 24–27, 1986.

[2] N.A. Kheir, K.J. Åström, D. Auslander, K.C. Cheok, G.F. Franklin, M. Masten, and M. Rabins, "Control systems engineering education," *Automatica*, vol. 32, no. 2, pp. 147–166, 1996.

[3] R.M. Murray, K.J. Åström, S.P. Boyd, R.W. Brockett, and G. Stein, "Future directions in control in an information rich world," *IEEE Contr. Syst. Mag.*, vol. 23, no. 2, pp. 20–33, 2003.

[4] K.J. Åström and A.-B. Östberg, "A teaching laboratory for process control," *IEEE Contr. Syst. Mag.*, vol. 6, no. 5, pp. 37–42, 1986.

[5] K.J. Åström and M. Lundh, "Lund control program combines theory with hands-on experience," *IEEE Contr. Syst. Mag.*, vol. 12, no. 3, pp. 22–30, 1992.

[6] Quanser Consulting Inc. *Quanser Engineering DC Motor Control Trainer User Manual*, 2004.

Jacob Apkarian (jacob.apkarian@quanser.com) received the B.Eng. from McGill University in 1980 and the Ph.D. from the University of Toronto in 1989. He was an assistant professor at the University of British Columbia from 1990–1992 and a senior controls engineer at SPAR Aerospace from 1992–1996. He founded Quanser Consulting in 1990. He can be contacted at Quanser, 80 Esna Park Drive, Unit 1, Markham, Ontario, L3R 2R6.

Karl Åström received the Ph.D. in 1960 from the Royal Institute of Technology, Stockholm, Sweden. He was employed by IBM Research from 1960–1965, and he was a professor of automatic control at the Lund Institute of Technology from 1965–2000. He received the IEEE Medal of Honor 1993.