Using MATLAB in Graduate Electrical Engineering Courses

Asad Azemi, Christopher Stook
Department of Electrical Engineering
Penn State University
Great Valley Campus
Malvern, PA 19355

Abstract. Control system design packages like MATLAB, MATRIX_X, Control C, SIMNON, etc. have become essential ingredients of both undergraduate and graduate courses in the systems and controls area. This work describes our experience, at the Great Valley Campus of the Pennsylvania State University, with the use of one of these packages, namely MATLAB with its Control Systems, Signal Processing, and Symbolic Math Toolboxes in Optimal Control and Linear Control Systems courses. This paper will also present the **MATLAB** features that are found most useful in these two courses in enhancing students' understanding of the material. The emphasis will be on presenting some of the features of the MATLAB which can be used to produce interactive simulations, using user interface controls, and analytical solutions. Examples illustrating these features will be included and the actual code will be posted at a site on the World Wide Web. The address of the site will be given during the conference. A discussion of how MATLAB helps in reducing the amount of time spent in performing computational homework assignments will follow. Finally, the general student reaction to incorporating this software package into these courses will be reported.

Introduction

Penn State Great Valley Campus, one of the eighteen campuses of the Penn State University, is a graduate center designed to address the educational need of the working engineers in Philadelphia area. Almost all of our students are working engineers, with a wide variety of backgrounds using simulation packages. The Electrical Engineering Department at the Pennsylvania State University-Great Valley Campus is incorporating computer aided engineering (CAE) packages into its curriculum. The intent of augmenting the curriculum with these packages is to enhance the students' theoretical understanding of the material with hands on analysis and design experience. The benefits of CAE packages in the classroom have been realized by the authors and their co-workers before [1-5]. The benefits of using these packages in a university setting is also confirmed by the number of new undergraduate textbooks, and revisions of previously printed textbooks incorporating new exercises and problems based on these packages, such as [6-13]. This trend can also be seen in recent graduate textbooks [14-15]. A summary of the advantages and disadvantages of incorporating these packages into our graduate curriculum are presented below.

General Advantages

One of the main advantages of using these tools is the reinforcement of student understanding of theoretical principles by means of enhanced graphical aids and interactive simulations. The simple graphical nature of the simulation outputs tend to help students understand the behavior of mathematically complex systems. This feature of the simulation packages could be used in preparing courseware products.

Another equally important advantage is the preparation for analysis and the design of more complex systems than those that can be treated with pencil and paper, much like the ones students will see in industry. With these tools, instructors can assign fairly complex design problems that otherwise would be unrealistic without the help of such software. This is a key advantage that helps students apply the theoretical principles learned in the classroom to the real world problems associated with following a design cycle through completion.

With the advancement of the CAE packages, it is no longer necessary to spend time working out all the mathematical details that are part of a problem but are not its main focus. For example, in finding the state transition matrix one needs to use the inverse Laplace transform, which is not the main focus of the problem. CAE packages can help students by performing some of these routine and time consuming mathematical steps, allowing more time to be spent on the new concepts.

Student response concerning the use of these packages is generally favorable. One interesting response received from students is an increased interest in the subject material. Student interest is heightened by the prospect of being able to design a fairly complicated system that would otherwise have to be studied only theoretically. Moreover, they were pleased with the time that was saved and the increased accuracy using these packages to perform routine mathematical steps. Some students are interested in using these packages in their professional careers.

General Disadvantages

Three of the disadvantages of using these packages are the maintenance and operation of these packages on an accessible computer system, the extra work required by students (and instructors) to learn how to use CAE packages, and assuring that the packages are included in the baseline curriculum as part of the required course material.

Requiring the use of these packages in homework assignments generally implies 24 hour student access to computer facilities. These packages tend to be expensive and student versions either do not exist or are inadequate for some problems.

The second disadvantage is the additional time requirements for both students and instructors. The additional time required for students to learn how to use CAE packages must be acknowledged and accounted for when assigning homework. The additional time required for the instructor includes not only the time it takes to develop and run through each simulation, but also the time required to generate handouts detailing the steps required to log into the system, use the particular package, and obtain printouts of results. The instructor must come up with tutorials to compensate for the lack of textbooks available on the use of these packages in specific areas that the courses cover. This task is made more difficult by the lack of good documentation accompanying the packages. Furthermore this extra effort can not easily be quantified to be of any use in tenure/promotion/merit raise decisions by the administrators.

The third disadvantage is in defining a baseline in the curriculum for teaching these software packages. As different instructors rotate through undergraduate teaching responsibilities, the degree of emphasis in using these packages will not be the same. This creates students with different knowledge levels. This result in frustration for the students who did not have adequate exposure to these material if assignments are made assuming previous familiarity with the software package. It should be mentioned, since almost all of our

students are working engineers, with a variety of backgrounds, this disadvantage is more significant in our situation.

Analytical Solutions via MATLAB

Previous works in using MATLAB [16] in graduate level control systems, such as [3-4], have been focusing on the numerical solutions and simulation capabilities of MATLAB. In this section we will introduce, by way of several examples, some functions in the Symbolic Math toolbox that are useful in obtaining analytical solutions to common control system problems.

A typical problem in linear control is to obtain the transfer function matrix of a system given in state space representation. If the system is given by

$$\dot{x}(t) = Ax(t) + Bu(t)$$
$$y(t) = Cx(t) + Du(t)$$

where A, B, C, and D are constant matrices of appropriate dimensions, the transfer function matrix, $H(s)$, will be given by

$$H(s) = C[sI - A]^{-1}B + D$$

In such a problem one can convert the system parameters to symbolic representation, using *sym()* function, and then computing the transfer function using *inverse()*, and *symop()* functions. The *inverse()* function will produce the inverse of a matrix, The *symop()* function allows to add, subtract, multiply, and divide several symbolic expressions. The final answer will be in terms of thes parameter. The working MATLAB code for this problem can be found under Example 1 at our Web site.

Another common problem in linear system is obtaining discrete-time representation of a system from its continuous-time model using a specified sampling period. By using Symbolic Math functions one can obtain an analytical representation of the system parameters in terms of the sampling period. This may provide more insights about the system behavior. In such a problem one can use the *sym()*, *symop()* and *invlaplace()* functions. The *invlaplace()* function will calculate the inverse Laplace transform. Care has to be taken in using this function since, it will only work with scalar functions. The working MATLAB code for this problem can be found under Example 2 at our Web site.

Another typical problem is to find the solution of the linear nonhomogenous state equation by using state transition matrix. If a system is given by

$$\dot{x}(t) = Ax(t) + Bu(t)$$

the solution of the state equation is obtained by

$$x(t) = L^{-1}[\Phi(s)]x(0) + L^{-1}[\Phi(s)Bu(s)]$$

where L–l indicates the inverse Laplace transform and $\Phi(s) = (sI - A)^{-1}$ represents the state transition matrix. The solution can also be presented in terms of $\phi(t)$ and convolution integral as:

$$x(t) = \phi(t)x(0) + \int_0^t \phi(\tau)Bu(t-\tau)d\tau$$

$\phi(t)$ can be calculated by taking the inverse Laplace transform of $\Phi(s)$ or by using Cayley-Hamilton method and the eigenvalues of the system matrix A. *In* addition to the functions that where mentioned in previous examples, the function *solve()* has been used to obtain the state transition matrix. This function solves simultaneous system of equations. Example 3, posted at our Web site, demonstrates both techniques mentioned above to find the state transition matrix.

1996 ASEE Annual Conference Proceedings

It should be mentioned that the use of Symbolic Math toolbox may not always be appropriate. In some cases the complexity of the answer obscures the insight that expected from an analytical solution.

Interactive Simulations with MATLAB

MATLAB is also capable of producing interactive simulations. This can enhance the quality of presentation with graphical simulations. With the help of interactive simulations instructors can effectively illustrate the change in system response due to parameter variations. This helps students gain a better understanding of the subject. Moreover, since there is no need for students to do any programming, this will allow students with limited or no knowledge of MATLAB programming to access features of MATLAB with little investments of time. This feature is essential in an interactive courseware development.

Interactive simulations are produced with the help of graphical user interface (GUI) functions. The GUI is made up of graphical objects, such as menus, buttons, lists, and fields. These objects have meanings; when a user "chooses" an object there is an expectation that a certain kind of action will take place. In MATLAB the GUI is implemented using user interface (UI) controls.

Figure 1 shows the application of user interface in minimum control energy for a scalar system with fixed final state. The system model and performance index are given by

$$x_{k+1} = a x_k + b u_k$$
$$J_0 = \frac{r}{2} \sum_{k=0}^{n-1} u_k^2$$

where x_k and u_k denote the state and control input, respectively. r is a positive scalar weighting factor.
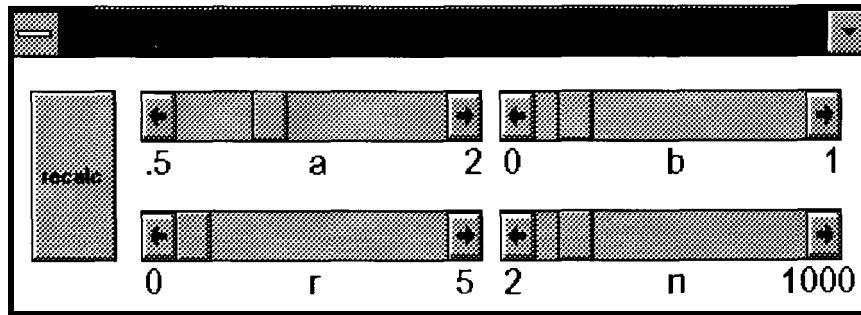

Figure 1. User interface control for minimum control energy problem.

This program will produce plots of the state and the optimal input vs. time. The sliders will allow the user to change the system parameters and observe the resulting changes.

Figure 2 illustrates the use of user interface control in a discrete-time fixed final state linear quadratic regulator. The system model and performance index are given by

$$x_{k+1} = A x_k + B u_k, \qquad\qquad k > i$$
$$J_i = \frac{1}{2} x_N^T S_N x_N + \frac{1}{2} \sum_{k=i}^{N-1} \left(x_k^T Q x_k + u_k^T R u_k \right), \qquad S_N \geq 0,\, Q \geq 0,\, R > 0$$

and the optimal control law equations given by

$$K_k = \left(B^T S_{k+1} B + R\right)^{-1} B^T S_{k+1} A, \qquad\qquad S_N \text{ given}$$

$$S_k = A^T S_{k+1}(A - BK_k) + Q$$

$$V_K = (A - BK_k)^T V_{k+1}, \qquad\qquad\qquad V_N = C^T$$

$$P_k = P_{k+1} - V_{k+1}^T B\left(B^T S_{k+1} B + R\right)^{-1} B^T V_{k+1}, \qquad P_N = 0$$

$$K_k^u = \left(B^T S_{k+1} B + R\right)^{-1} B^T$$

$$u_k = -\left(K_k - K_k^u V_{k+1} P_k^{-1} V_k^T\right) x_k - K_k^u V_{k+1} P_k^{-1} r_N, \qquad r_N \text{ given}$$

The system parameters are specified in continuous time and are discretized based on the sampling period T. The program produces plots of output and control input for optimal and suboptimal regulator. The value of T and the final time, $t(f)$, can be changed with the slider bars. The "A", "B", and "C" buttons allow the user to change the system parameters. The "system" button prints system parameters in continuous- and discrete-time, sampling period, poles of the system, and the final time. The "recalc" button updates the plots based on the parameter changes.
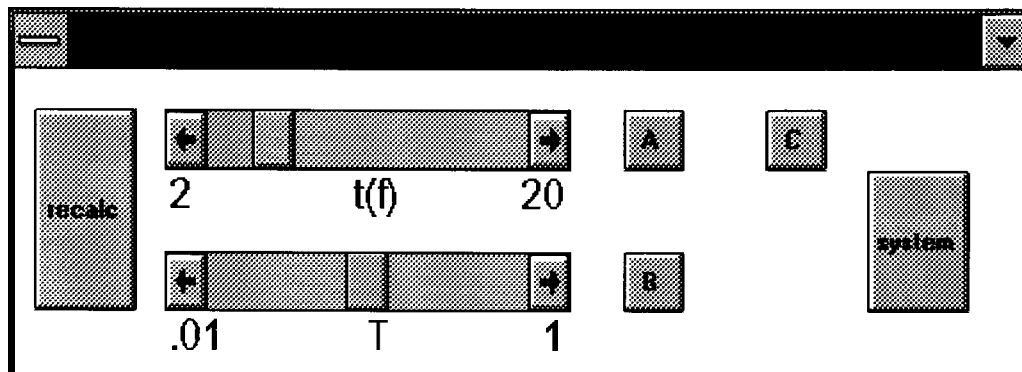

Figure 2. User interface control for fixed final state LQ regulator.

Although, the GUI is very useful for presenting the changes in system response due to parameter variations, the implementation is very time consuming and requires an extensive knowledge of MATLAB.

Conclusion

In this paper, we have presented the use of MATLAB software packages in our graduate curriculum, at Penn State University. Several of the advantages provided by computer simulation packages, such as MATLAB, include a reinforcement of student understanding of theoretical principles, by means of graphical representations and interactive simulations, allowing assignment of larger and more complex designs, increased student attentiveness, and allowing more time to be spent on new materials, by performing routine and time consuming mathematical operations. The main disadvantages of using computer simulation packages are the extra work required of students and instructors, assuring that the packages are inserted in the baseline curriculum as part of the required course material. The general student reaction to the use of MATLAB has been positive. Due to the variety of student backgrounds, some students have had no prior exposure to the

MATLAB which made it difficult for them to finish all the assignments. Using user interface controls will reduce problem but at the same time will place more burden on the instructor. Although, the students have access to MATLAB and its Control Systems and Signal Processing toolboxes in our computer lab, most of them prefer to purchase the "Student Edition of the MATLAB, Windows Edition." This gives them an opportunity to work with MATLAB at their own convenience. But, it should be also mentioned that the student version is a subset of the MATLAB toolboxes, and not all the functions are available.

References

[1]   T.W. Martin, A. Azemi, D. Hewett, and C.P. Schneider. *PSpice in Electrical Engineering Laboratories.* Proceedings of the ASEE Annual Conference, pp. 1307-1308, 1992.

[2]   D. Andrews, A. Azemi, S. Charlton, and E. Yaz. *Computer Simulation in Electrical Engineering Education.* Proceedings of the ASEE Gulf-Southwest Section Meeting, pp. 77-82, 1994.

[3]   A. Azemi and E. Yaz. *PSpice and MATLAB in Undergraduate and Graduate Electrical Engineering Courses.* Proceedings of the 24th Frontiers in Education Conference, pp. 456-459, 1994.

[4]   E. Yaz and A. Azemi. *Utilizing MATLAB in two Graduate Electrical Engineering Courses.* Proceedings of the 25th Frontiers in Education Conference, pp. 2c6.1-2c6.4, 1995.

[5]   Roger L. Haggard. *Classroom Experiences and Student Attitudes toward Electronic Design Automation.* Proceeding of the 25th Southeastern Symposium on Systems Theory. IEEE Computer Society Press, pp. 411-415, 1993.

[6]   R. Dorf, and R. Bishop. *Modern Control Systems,* seventh edition, Addison-Wesley Publishing Company, 1995.

[7]   G. Franklin, J. Powell, and M. Workman, *Digital Control of-Dynamic Systems.* Addison-Wesley Publishing Company, 1990.

[8]   R. Strum, and D. Kirk. *Contemporary Linear Systems using MATLAB.* PWS publishing company. 1994.

[9]   H. Saadat. *Computational Aids in Control Systems Using MATLAB.* McGraw-Hill, 1993.

[10]   B. Shahian, and M. Hassul. *Control System Design Using MATLAB.* Prentice Hall, N. J., 1993.

[11]   N.E. Leonard, and W.S. Levine. *Using MATLAB to Analyze and Design Control Systems.* Benjamin/Cummings, 1992.

[12]   K. Ogata. *Solving Control Engineering Problems with MATLAB.* Prentice Hall, N. J., 1994.

[13]   D.C. Hanselman, and B.C. Kuo. *MATLAB Tools For Control System Analysis and Design.* Prentice Hall, 2nd edition, 1995.

[14] Frank L. Lewis, and Valssilis L. Syrmos. *Optimal Control.* John Wiley& Sons, 2nd edition, 1995.

[15] Dorato, *et al. Linear-Quadratic Control An Introduction.* Prentice Hall, 1995.

[16] The MathWorks Inc., 21 Eliot St., South Natick, MA, 01760.

ASAD AZEMI

Asad Azemi is an assistant professor of Engineering at the Pennsylvania State University. He received his B.S. degree from University of California at Los Angeles in 1982, M.S. degree form Loyola Marymount University in 1985, and Ph.D. degree from University of Arkansas in 1991. His professional interests are in control theory, signal processing and the use of computers in undergraduate and graduate education.

CHRISTOPHER STOOK

Christopher Stook received his B.S. degree in electrical engineering from the Pennsylvania State University in 1992. He is currently working at Phoenix Microwave Corp. and is working towards his masters degree at the Pennsylvania State University.