# The Effective Use of Simulation in the Introductory Controls Curriculum

## Abstract

In this paper, the use of simulation in the introductory controls curriculum is explored as a means for teaching students about the limitations of the linear analysis techniques they have been taught, as well as for teaching students how to address these limitations. In particular, this work proposes specific techniques for introducing the use of SIMULINK and such advanced topics as nonlinear systems, time-varying/hybrid systems, optimality, robustness, multi-input-multi-output systems, and digital control systems in an introductory course. The approaches proposed in the paper attempt to build intuition without requiring significant amounts of in-class instruction.

## Overview

The use of computer tools in introductory controls courses has become almost ubiquitous. This statement is exemplified by the inclusion of material on the instruction of MATLAB in most any introductory controls textbook[1,2,3,4]. These computer tools allow students to perform with a single command what would otherwise take pages of hand calculations. The power of these tools has been leveraged primarily to allow students to solve a greater number of problems and to solve problems that have a level of complexity that make them prohibitively difficult to solve by hand. These advantages are significant and allow students to build intuition and focus more on broad concepts rather than mathematical details. For example, students can build intuition by using MATLAB to experiment with the systems and controllers they are designing[5]. A student can generate a series of time response curves for a system with a particular parameter, such as damping ratio, varied over a range of values. Similarly, a student can vary the values of a controller gain over a range of values to understand its impact on performance. Exercises such as these can even be performed during lecture to help students connect to the material. In this paper, it is proposed to expand the use of these computer tools to place a greater emphasis on simulation in order to help students better understand the applicability of the theory they are learning, as well as to teach them how to overcome the limitations of the theory.

Introductory controls courses generally introduce techniques for analyzing and designing the control for continuous, linear time-invariant (LTI), single-input-single-output (SISO) systems. Since these techniques tend to be somewhat complicated mathematically, it is natural for students to focus on details such as how to compute the exact location of the break-out point in a root locus plot. As mentioned above, the use of computer tools like MATLAB can mitigate this problem by allowing students the time to experiment. Even with this improved intuition, students often complete these courses without recognizing that most systems they will encounter in the real world are inherently nonlinear, time-varying, have multiple inputs and multiple outputs, and that controllers are often implemented digitally. Furthermore, students rarely complete these courses with an understanding of which systems the theory they have learned applies, as well as how to address the complexities that are beyond the scope of the theory they have been taught. Even when simulations and laboratory exercises are emphasized in a course, they are often employed to illustrate the LTI SISO theoretical concepts learned in the class, rather than the exceptions to the theory[6].

Most instructors likely would agree with the importance of teaching these more advanced concepts, but struggle with how to fit them into an already crowded curriculum. It is proposed here that the use of simulation can help students better understand the types of systems their theory applies to without adding significantly to the amount of material that must be covered in class. The use of simulation also provides students a powerful tool for addressing the complexities of the real world that are beyond the scope of the theory that they have been taught. Even if students do not learn how to completely address these complexities, it is important that they are aware that these subtleties exist so that as industrially practicing engineers they can seek further information if necessary. Tools such as SIMULINK, Saber, and MATRIXx are commercially available for simulating closed-loop feedback systems. In this paper, we specifically present techniques using the SIMULINK addition to MATLAB since it is the tool most widely employed in academia.

In order to minimize the impact on the current course schedule, it is suggested that students learn these lessons primarily in homework assignments. For example, as an addition to a standard homework problem, the students could be required to simulate the system with the addition of some complexity that they were not able to address analytically. The students may further be required to comment on the effect of this additional complexity. In order to make sure that students are not overburdened by this additional element, they can be given a picture of the necessary SIMULINK block diagram or even a copy of the actual SIMULINK file.

## Introducing SIMULINK

If students do not have prior experience with SIMULINK, the topic can be introduced into a course without investing significant class time. Specifically, the SIMULINK toolbox can be introduced in about 15 minutes. This time is spent demonstrating how to open the software, create a new model, drag blocks from the library, connect the blocks, and run a simple simulation[7]. In particular, it is useful to show source blocks like a step input or clock, math blocks like a gain or transfer function, and sink blocks like an XY graph or an output to the workspace. It is also useful to take a minute to demonstrate how to change the configuration parameters of the model like the simulation run length and solver step size. Students should also be warned about numerical difficulties that can introduce errors into their simulation results. Finally, it is necessary to demonstrate how to generate the graphs that the students will be required to turn in for their homework assignments.

This introduction is by no means complete, but it gets the student far enough that they may begin to explore the software on their own. To facilitate this process, assigning a homework problem early in the course that requires students to recreate and run a block diagram that they have been given a picture of, such as the one shown in Figure 1, is recommended. Within the problem statement, particular settings that the students may not know or remember may be identified. For example, the statement, “make sure to change the settings of your ‘To Workspace’ blocks to have Array outputs (not Structure outputs)” could be included. Another means to facilitate students’ learning of SIMULINK is to refer them to a textbook[8] or website tutorial[7]. Of the many standard introductory controls textbooks available, Nise offers significant coverage of SIMULINK[3].

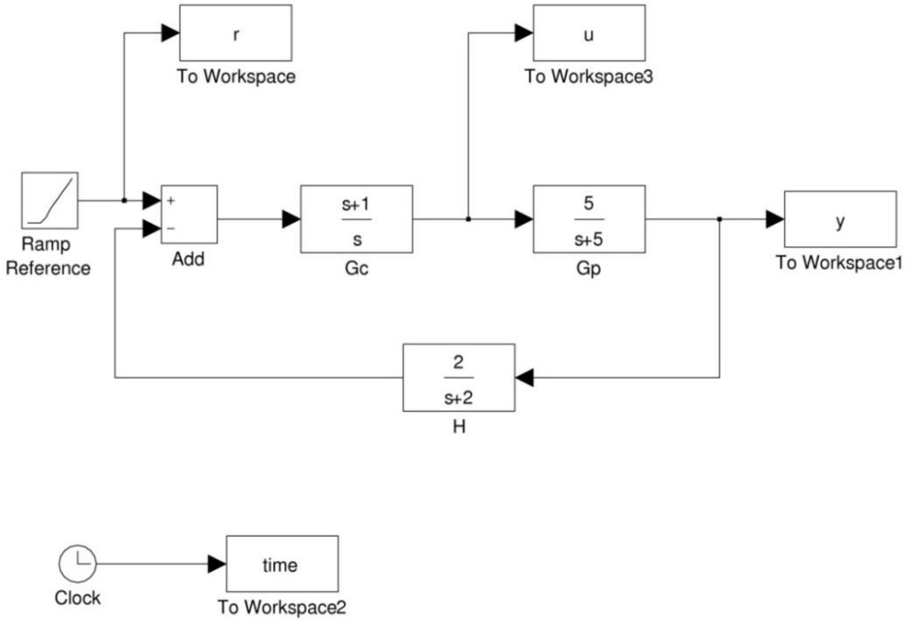

Figure 1 Introductory SIMULINK diagram

## Nonlinear Systems

The first advanced topic that will now be addressed is systems with nonlinear dynamics. Most students who have taken an introductory control systems course have likely heard the statement, “All real systems are inherently nonlinear.” Of course, in many cases the nonlinearities that are present may be neglected or are not central to the issue being analyzed. Therefore, it is important for students to understand the implications of the nonlinearities and to be able to identify when it is reasonable to neglect them.

First, students need to understand that many of the techniques they have learned, such as transfer function representations and root locus techniques, do not apply directly to nonlinear systems. A common approach for analyzing and designing the control for nonlinear systems is to employ a linear approximation. Many students are familiar with the notion of linearizing a nonlinear function from their calculus sequence; that is, by taking the first terms of the Taylor series expansion of a nonlinear function. With dynamic systems, it is necessary to take this a step further and be able to linearize nonlinear differential equations about an equilibrium point[3]. If the topic of linearization is covered in the current or prerequisite courses, then simulation can be employed to help demonstrate its limitations as a technique for dealing with nonlinear systems. Specifically, depending on the severity of the nonlinearity, the analysis and design done for the linear approximation will break down as the system state moves away from the equilibrium point about which the linearization was performed. This lesson is taught by adding a simulation component to a standard problem where the students generate a linear approximation for a nonlinear model. The simulation component requires the students to simulate the original nonlinear system for some input and compare the output to the output obtained from a simulation of the linear approximation of the system. Students may have difficulty generating the SIMULINK model for the nonlinear system, but this can be addressed by providing them the SIMULINK diagram or file.

As an example, consider the mass-spring-damper system with nonlinear spring pictured in Figure 2. The value of the damping constant is 3 N·s/m and the mass M is 5 kg, while the force in Newtons generated by the spring is given by the nonlinear function $f = -1.1x - x^3$.

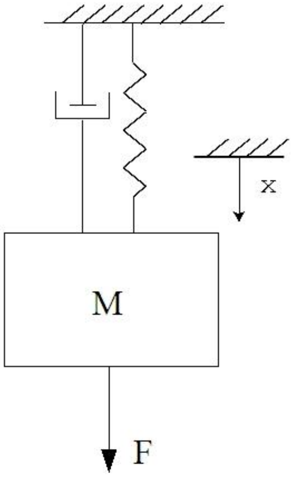

Figure 2 Example mass-spring-damper system with nonlinear spring

The nonlinear differential equation describing the behavior of this system is, therefore,

$$F = 5\ddot{x} + 3\dot{x} + 1.1x + x^3$$

A typical question that could be asked would be to find a linear approximation of this differential equation in terms of the deviations $\delta F$ and $\delta x$ from the equilibrium for some nominal force $F_{nom}$. This linear approximation then could be represented as a transfer function where $\delta F$ is the input and $\delta x$ is the output. The SIMULINK models given in Figure 3 and Figure 4 then could be employed to assess the effect of the nonlinearity for a sinusoidal forcing input.

Even if the introductory controls curriculum does not address linearization, students still can investigate nonlinear systems by introducing nonlinearities like saturation, dead zone, and backlash using blocks from the SIMULINK library[3]. Otherwise, students can employ the MATLAB command `linmod` to generate the linear approximation as a state space model. If there is time, students can investigate the implications a nonlinear system has for properties like stability[9], superposition, and homogeneity.

## Time Varying/Hybrid Systems

Another type of system not normally addressed in the introductory controls curriculum is a time-varying system. Time-varying systems can be explored by using blocks from the SIMULINK math library to calculate a quantity based on the current simulation time. A time-varying example could be a rocket ship whose mass changes as fuel is rapidly burned during lift-off.

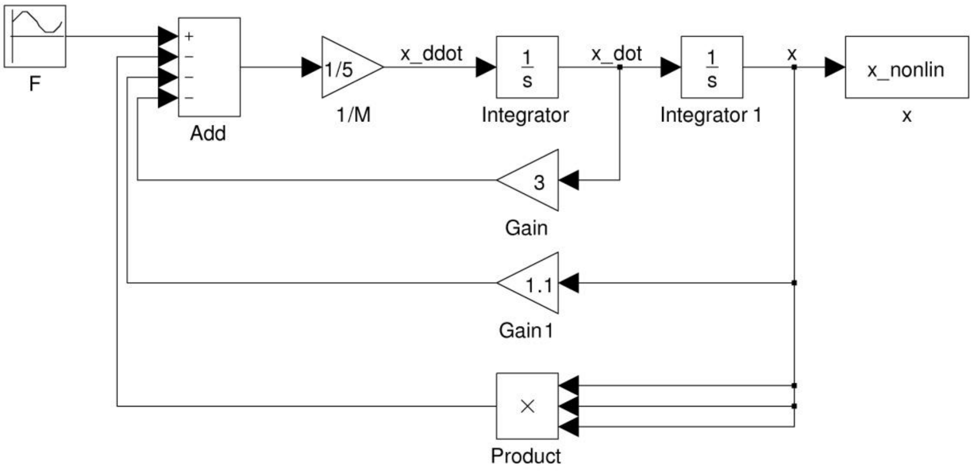

**Figure 3 SIMULINK model of the nonlinear mass-spring-damper system**

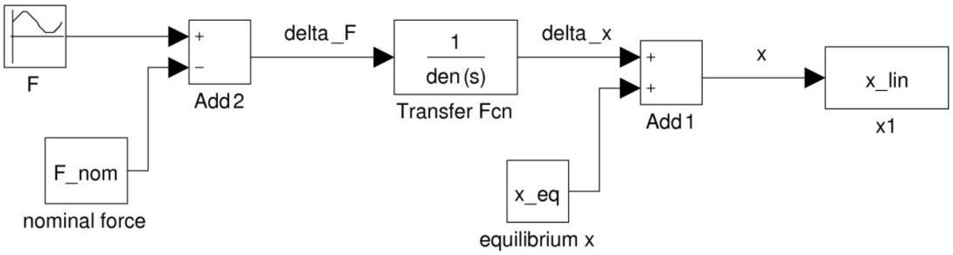

**Figure 4 SIMULINK model of the linearized system**

A specific type of time-varying system is a hybrid system, that is, a system that has both continuous and discrete aspects. An example of a hybrid system is an automobile that switches between a gasoline engine and an electric motor for propulsion. Another common example is a control system that employs gain scheduling. Here the controller gains change discretely based on some internal or external triggering event. A simple hybrid system can be simulated using a switch or multi-port switch block in SIMULINK.

Students should learn from these examples to determine when a time-varying model is necessary based on the speed and magnitude of the variation. For example, considering the rocket mentioned above, during lift-off the mass changes very quickly when compared to the dynamics of the rocket, whereas during re-entry less fuel is consumed and the mass changes at rate much slower than the dynamics of the rocket. Therefore, during lift-off a time-varying model is justified, while during re-entry a time-invariant model is sufficient.

An especially powerful example involves switching between two linear models that are individually stable in order to produce an output that grows unbounded. This can be achieved by considering a system with two states such that the closed-loop poles are placed to achieve a stable spiral trajectory. By alternating which of the two closed-loop poles are faster, the eccentricity of the spiral can be switched between the two principal directions. Switching between these two trajectories at a particular rate can lead to an unstable response. Such an example could arise in a gain scheduled control system for instance.

The SIMULINK diagram for an example hybrid system[10] is shown in Figure 5.  This example is based on the satellite system shown below[2]. In this system the inputs $u_1$ and $u_2$ are the radial and tangential thrusts, the state variables $x_1$ and $x_3$ are the radial and angular deviations from the reference orbit, and the outputs $y_1$ and $y_2$ are the radial and angular measurements, respectively.

$$\begin{bmatrix} \dot{x_1} \\ \dot{x_2} \\ \dot{x_3} \\ \dot{x_4} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 3\omega^2 & 0 & 0 & 2\omega \\ 0 & 0 & 0 & 1 \\ 0 & -2\omega & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}$$

For this example, it is assumed that $\omega = 2\pi$ /(3600*24) rad/sec corresponding to an orbit synchronous with the earth's rotation. A state-feedback approach to control is used where the gain K1 places the four closed-loop poles at −0.02 ± 0.4i and −0.004 ± 0.4i, and the gain K2 places the poles at −0.004 ± 0.35i and −0.02 ± 0.4i. The pulse generator of Figure 5 is then set to have a period of 8.3 seconds and the threshold of the switch is set to 0.5. The state-space block is defined such that its C matrix is the identity matrix so that the controller may have access to the full state vector. The C matrix outside of the state-space block is set to [1 0 0 0] in order to observe the radial position. For these parameters, the radial position will be observed to grow unbounded in the switched system even though the closed-loop system is stable for each of the individual feedback gains. Despite the fact that this example is contrived, and perhaps not very realistic, it is a powerful lesson.

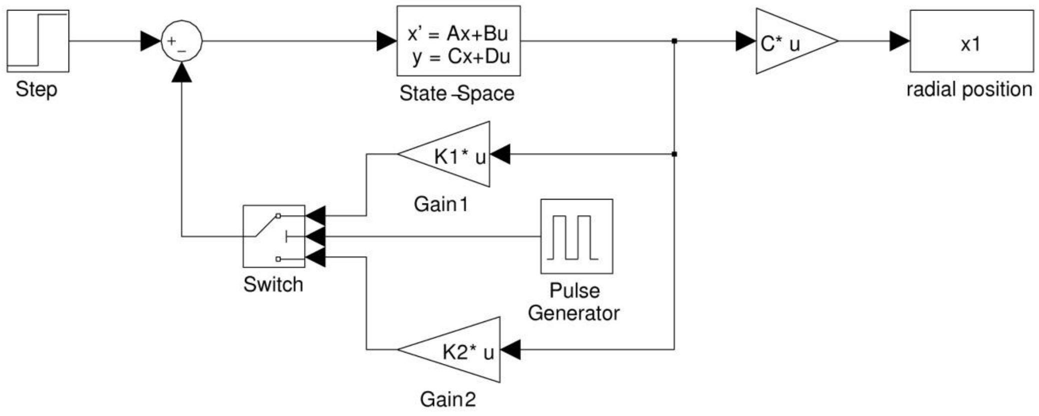

Figure 5 Hybrid system example

For hybrid systems with more complicated discrete dynamics, the Stateflow toolbox of SIMULINK can be employed. In some courses, such as a course on manufacturing control, the discrete-event aspect of the systems could be explored more thoroughly[10].

## Optimality

Optimal control techniques are widely employed in industry, yet the concept of optimality generally is not considered in the undergraduate controls curriculum. In most instances, when undergraduates are asked to assess the design of a controller, their conclusions are based solely on the resulting tracking error, or more specifically, on the transient and steady-state error. With more advanced control techniques, optimality is sometimes defined solely on the basis of the mean-squared error, but more commonly it also includes the amount of control effort employed. Control effort is important because it generally reflects the cost of fuel or power needed for achieving the prescribed control. It can also reflect the size of the actuator needed.

The subject of optimality can, therefore, be added to a class by simply requiring students to add an output to their simulations for control effort. An example of this can be seen in the SIMULINK diagram shown in Figure 1. The subject of optimality is especially useful for comparing controller designs. For example, a closed-loop system requiring a certain overshoot and level of steady-state error can be achieved with either a lead controller or a lag controller. The choice between controllers then must be made based on the trade-off between transient response and control effort. In general, the lead controller will provide faster response at the expense of larger control effort. This is also a good opportunity to remind students that all systems are inherently nonlinear. In particular, no real actuator can provide infinite control effort; at some point the actuator will saturate. The effect of saturation can be modeled quite simply using a saturation block in SIMULINK.

If the course being taught includes material on state-space approaches to control (and estimation), then the concept of a cost function in the context of Linear Quadratic Regulation can be introduced as a means for achieving "optimal" control. Furthermore, MATLAB commands can be employed for calculating the optimal state-feedback gains, without much instruction on the mathematical details being necessary.

## Robustness

Another aspect of control systems that is widely considered in industrial applications is a system's robustness properties. Robustness is taught in the undergraduate curriculum in terms of stability margins as part of the instruction on frequency response techniques. This again is a situation where students do not always fully grasp the physical importance of the mathematical quantities of phase and gain margin; rather, they just go through the motions of calculating the correct numerical answer. Simulation is a means by which students can build intuition. Specifically, students could design or be given a closed-loop system with small stability margins. They then could simulate this system for a plant with slightly different parameters, reflecting modeling uncertainty or unmodeled dynamics, such that the resulting closed-loop behavior is unstable. Students could also simulate the system with additional exogenous inputs like disturbances and noise[9], see Figure 6. Even if the system remains stable, the students could be asked to compare the tracking properties of the simulation with the "ideal" model and the "real" model.

The above examples illustrate how robustness properties can be investigated without delving into more advanced topics like sensitivity and complementary sensitivity functions[1,2]. Even without the additional instruction, students can still investigate the tradeoffs between a system's response under the influence of disturbances and noise through simulation. For example, students can be required to simulate a system first with a step disturbance, then with sinuisoidal noise. The students can then increase the controller gain and observe that the new system was better able to reject the disturbance, but at the expense of amplifying the high frequency noise. Students can also investigate this result mathematically by finding the transfer functions from the disturbance to the output and from the noise to the output. This is a skill that the students certainly will have been taught. This also presents a good opportunity to introduce the property of superposition and segues nicely into multi-input-multi-output (MIMO) systems.

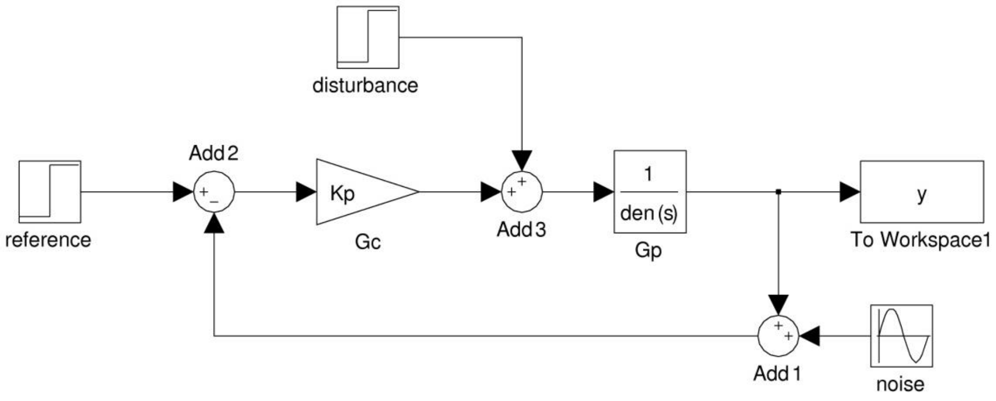

Figure 6 SIMULINK model with multiple exogenous inputs

## Multi-Input Multi-Output Systems

In the above discussion on robustness, systems with multiple inputs and a single output were introduced. If time permits in a course, students may then investigate systems with multiple inputs and multiple outputs. In courses that teach state-space design methods like state-feedback control, the extension to MIMO systems is trivial. The problem with these state-space methods is that they are not very intuitive and often not as informative as the graphical methods the students had learned for SISO systems. For example, in an undergraduate course students likely will not learn how to assess stability margins for a state-space system. In terms of design, undergraduate students are further not likely to be taught how to determine the placement of zeros for a state-space representation or how to implement something as fundamental as integral control action.

In many practical instances, these MIMO systems can be approximated as a series of SISO systems. For example, the attitude control of a satellite can be approximated as three SISO systems (one for each of three dimensions) if the product of inertia terms for the satellite are small compared to the principal moments of inertia. Similarly, the longitudinal and transverse motion of an automobile can be treated independently if the road being traveled upon has only curves with large radii of curvature. Therefore, the students may use simulation to assess the error introduced by approximating a MIMO system as a series of SISO systems.

For example, consider the MIMO satellite system introduced in the section on hybrid systems. The differential equations representing the system are repeated below where $x_1(t)$ and $x_3(t)$ represent the radial and tangential positions of the satellite respectively.

$$\ddot{x}_1 = 3\omega^2 x_1 + 2\omega\dot{x}_3 + u_1$$
$$\ddot{x}_3 = -2\omega\dot{x}_1 + u_2$$

The students may be asked to take the Laplace transform of the above equations and perform the necessary algebraic manipulations to put the system equations into the following form.

$$\mathrm{X}_1(s) = G_{11}(s)\mathrm{U}_1(s) + G_{12}(s)\mathrm{U}_2(s)$$
$$\mathrm{X}_3(s) = G_{21}(s)\mathrm{U}_1(s) + G_{22}(s)\mathrm{U}_2(s)$$

In this form, the students can perform an analysis or design by disregarding the cross-coupling terms. They can then assess the approximation by simulating the full MIMO system. The SIMULINK diagram shown in Figure 7 can be provided to the students as an example.

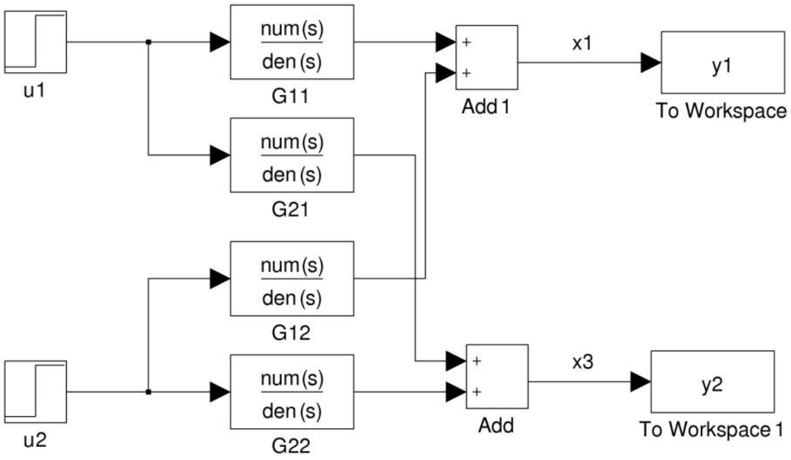

Figure 7 Example MIMO system with cross coupling

## Digital Control Systems

Digital control is a topic that is often included at the end of introductory controls textbooks[1,2,3], yet the material is rarely covered due to lack of time. Even though there is rarely room in the curriculum to cover material on the analysis and design of control systems in the discrete-time domain, simulation again can be employed to give students an understanding of digital control implementations without significantly impacting a course timeline. This can be achieved by adding zero-order hold blocks from the SIMULINK library to a simulation[7] as shown in Figure 8. With this example, students can see the effect of sampling in a control system. In particular, it is the goal to demonstrate to students that a digital implementation of their continuous controller design will behave as if there is a pure-time delay in their block diagram approximately equal to half the sampling time employed for each zero-order hold. Students may then draw the conclusion that a model that accounts for the sampling is necessary if the sample time is significant when compared to the speed of the system. It may also be possible to demonstrate to students the phenomenon of aliasing[9].

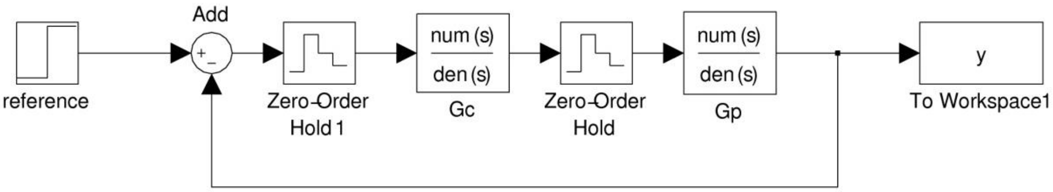

Figure 8 Example digital implementation in SIMULINK

## Summary

This work discusses the use of simulation in the introductory controls curriculum as a means for teaching students about the limitations of the linear analysis techniques they have learned as well as for teaching them how to address these limitations. In particular, it covers specific techniques for introducing the use of SIMULINK and for teaching such advanced topics as nonlinear systems, time-varying/hybrid/discrete systems, optimality, robustness, MIMO systems, and digital control systems in an introductory course. In any course, it is always a struggle to choose which topics to cover in the finite amount of time available. As such, it is nearly impossible to include all the techniques prescribed here in a single class. However, as many of the suggestions can be implemented with little class time needed, it is the hope that faculty will find the suggestions useful.

## Acknowledments

The author would like to acknowledge Tim Gordon, Peter Seiler, and Dawn Tilbury for their insights on control instruction that have helped to shape the content of this paper.

### Bibliography

[1] Dorf, R.C., & R.H. Bishop. *Modern Control Systems*, Prentice-Hall, 2003.
[2] Franklin, G.F., Powell, J.D., & A. Emami-Naeini. *Feedback Control of Dynamic Systems*, McGraw-Hill, 2003.
[3] Nise, N.S. *Control Systems Engineering*, John Wiley & Sons, 2004.
[4] Ogata, K. *Modern Control Engineering*, Prentice-Hall, 2002.
[5] Rowland, J.R. "Streamlined Learning in a Control Systems Course," In *Proceedings of 35th ASEE/IEEE Frontiers in Education Conference*, Indianapolis, IN, 2005.
[6] Bissel, C.C. "Control Education: Time for Radical Change?," *IEEE Control Systems Magazine*, pp. 44-49, 1999.
[7] Tilbury, D., & W. Messner. http://www.engin.umich.edu/class/ctms/index.htm, 1998.
[8] Dabney, J.B., & Bishop, R.H. *Mastering Simulink*, Prentice-Hall, 2004.
[9] Dormido, S., Dormido-Canto, S., Dormido, R., & N. Duro. " The Role of Interactivity in Control Learning," *International Journal of Engineering Education*, 2005.
[10] Hill, R.C., & D. Tilbury. "Introducing Discrete Event Systems into an Undergraduate Controls Course," In *Proceedings of ASEE Annual Conference and Exposition*, Chicago, IL, 2006.