

A VIRTUAL IMPLEMENTATION OF A DYNAMIC SIGNAL ANALYZER USING SIMULINK

John M. Watkins
Department of Electrical and Computer Engineering
Wichita State University
1845 Fairmount
Wichita, KS 67260-0044
J.Watkins@IEEE.org

Introduction

The ability to understand and utilize the frequency response of a linear system is a critical building block in many undergraduate engineering disciplines. For example, undergraduate students in electrical engineering will often see the frequency response in courses on circuits, signals and systems, communications, and control systems. Yet despite this significant exposure to and need for the frequency response, many undergraduate students have limited intuition on what it means and represents.

As we have learned in other areas of engineering education, for students to understand a concept they need to work with it, either in simulation, or ideally in experimentation. Unfortunately, equipment for measuring the frequency response, such as a dynamic signal analyzer or frequency response analyzer, is often too expensive and complex for a typical undergraduate engineering lab. Furthermore, it does not allow to students to experiment in simulation, when an experimental lab is either not feasible or desired.

Consequently, we have developed a virtual dynamic signal analyzer in SIMULINK for measuring the frequency response. To utilize the dynamic signal analyzer, the user places a dynamic signal analyzer (DSA) block in their SIMULINK model file. The DSA block has two inputs: the input and output signals from the linear system. The DSA block has one output, a sinusoidal signal that is used to drive the linear system. This signal is swept through a series of frequencies. By measuring the frequency

response at discrete frequencies, it provides a frequency response measurement with a high signal-to-noise ratio. More importantly, it allows the students to see the input and output signals at the different frequencies as they are being measured. The input and output signals in the time domain and the magnitude and phase of the frequency response are plotted in real time.

To run the virtual DSA in simulation mode requires only SIMULINK. However, if combined with the Quanser WinCon software and hardware input/output board, it can be used for measuring the frequency response of experimental apparatus in the laboratory. If the linear system is mechanical, the students can compare the system's movement with the responses they are seeing on the screen.

Other authors have used SIMULINK to create a virtual DSA. Wang, Abramovitch, and Franklin created a virtual DSA for verifying measurements and models of linear and nonlinear systems. The virtual DSA they developed was for simulation purposes only and had no interface with real hardware.[1]

On the other hand, the DSA developed by Lilienkamp and Trumper was written specifically to work with the dSPACE controller board. Besides the analog to digital (A/D) and digital to analog (D/A) converters, which you also find on the Quanser input/output board, the dSPACE controller boards have an on board digital signal processor (DSP). The onboard DSP provides for faster processing, but at a higher cost than Quanser input/output board.[2]

This paper will begin with a discussion of how the virtual DSA can be used in the classroom and the laboratory. The details of how the DSA works will be discussed from a software and mathematical perspective. The DSA is available from the author upon request.

DSA Use in the Classroom

As mentioned earlier, the virtual DSA can be used in the simulation environment and the laboratory environment, provided that the proper equipment exists. Let us consider the simulation environment first as most engineering schools will be able to utilize this feature of the DSA. To run the DSA, the user must have the following packages from The Mathworks: MATLAB, Simulink, and the Control System Toolbox. The Control System Toolbox is utilized only to store the measured frequency response in the Frequency Response Data (FRD) model format. While the FRD model format is ideal for working with the frequency response data, it is not necessary and could be easily be eliminated from the code if the Control System Toolbox is not available.

Let's look at a simple simulation of a vehicle suspension system. The equations of motion are given as follows:

$$\begin{aligned} M\ddot{x}_1 &= -K_1(x_1 - x_2 - x_{10}) - D(\dot{x}_1 - \dot{x}_2) - Mg \\ m\ddot{x}_2 &= K_1(x_1 - x_2 - x_{10}) + D(\dot{x}_1 - \dot{x}_2) - K_2(x_2 - y_r - x_{20}) - mg \end{aligned} \quad (1)$$

where M is the mass of the vehicle, K_1 and D represents the stiffness and the damping in the suspension, g is the acceleration of gravity, m is the mass of the tires and axles, and K_2 represents the stiffness in the tires.[3] The distances are given as follows: y_r is the roadway height, x_1 is the height of M above y_r , x_2 is the height of m above y_r , and x_{10} and x_{20} are the rest lengths of stiffnesses 1 and 2. A Simulink model of (1) is shown in Figure 1.

If you would like the students to understand how the vehicle ride would change under different road conditions, it would make sense for you to require them to find the frequency response between the roadway height y_r and the vehicle

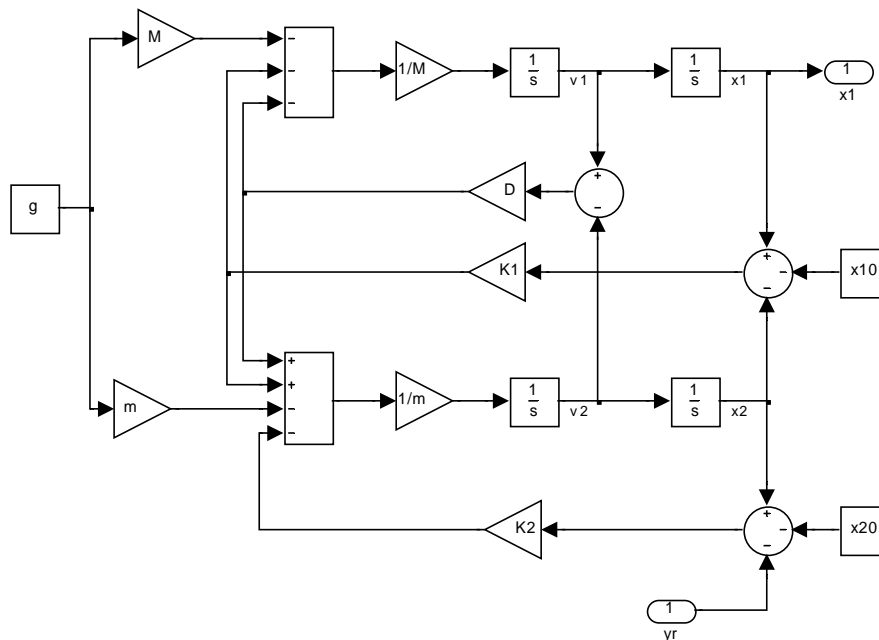


Figure 1: Simulink model of suspension system.

height x_1 . With some work, the students could find a linear model about the equilibrium point, enter this into MATLAB in the form of a state space or transfer function model, and calculate the frequency response using a MATLAB command such as Bode.

However, it might be more insightful for the students to “measure” the frequency response directly using the virtual DSA. To accomplish this, the suspension model shown in Figure 1, is placed in the suspension subsystem block shown in Figure 2. After the SIMULINK diagram is constructed, the user starts the DSA process with a MATLAB function developed for this block. The user supplies the function with the name of the SIMULINK model, the initial frequency, the final frequency, the number of frequency points, and an estimate of the system settling time. The function also needs to know whether you are running a simulation or a real-time experiment. Finally, there is an optional argument for the sampling time. If this argument is included, the program corrects for the phase delay introduced by sampling. When the DSA is used for system identification, this argument should be included. It should be left out when you are using the frequency response directly for digital control system design. The output of the DSA block in Figure 2 is a unit amplitude sinusoid that oscillates at the particular frequency specified by

the MATLAB function developed for this block. The DSA has two inputs, the input and output of the plant. Details of how the DSA works will be discussed later in the paper.

The suspension system was simulated using the following parameter values from[3]: $M = 300$ kg, $m = 50$ kg, $K_1 = 3000$ N/m, $K_2 = 30,000$ N/m, $D = 600$ N/ms⁻¹, $x_{10} = 1.5$ m, $x_{20} = 0.75$ m, and $g = 9.8$ m/s². The frequency response was measured at 20 logarithmically equally spaced frequencies between 1 (rad/sec) and 100 (rad/sec).

The DSA output after the first measurement at $\omega = 1.0$ rad/sec is shown in Figure 3. The display at the left shows one full period of the input signal (roadway height y_r) on Channel 1 and the output signal (vehicle height x_1) on Channel 2. At this frequency, the amount the vehicle moves is slightly larger than the amount the road moves. This gain shows up in the top right display with a magnitude of 1 dB. The movements of the vehicle and the roadway height are almost in sync. Consequently, the phase shift, which is shown in the lower right display, is approximately minus one degree.

However, as the students continued to watch the DSA run, they would see the display shown in Figure 4 for $\omega = 2.6$ rad/sec. Here the ride is

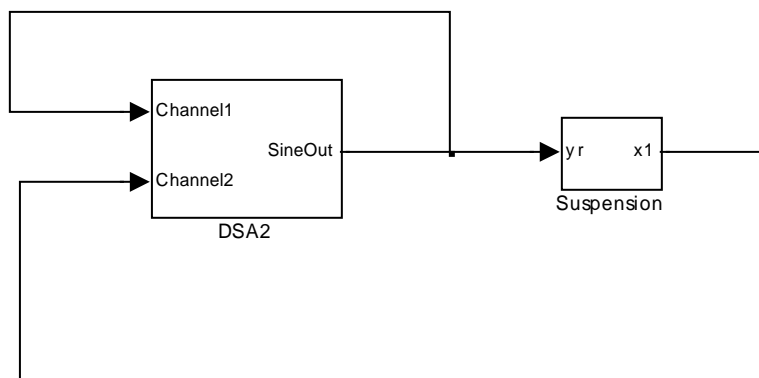


Figure 2: Simulink model of suspension system with dynamic signal analyzer.

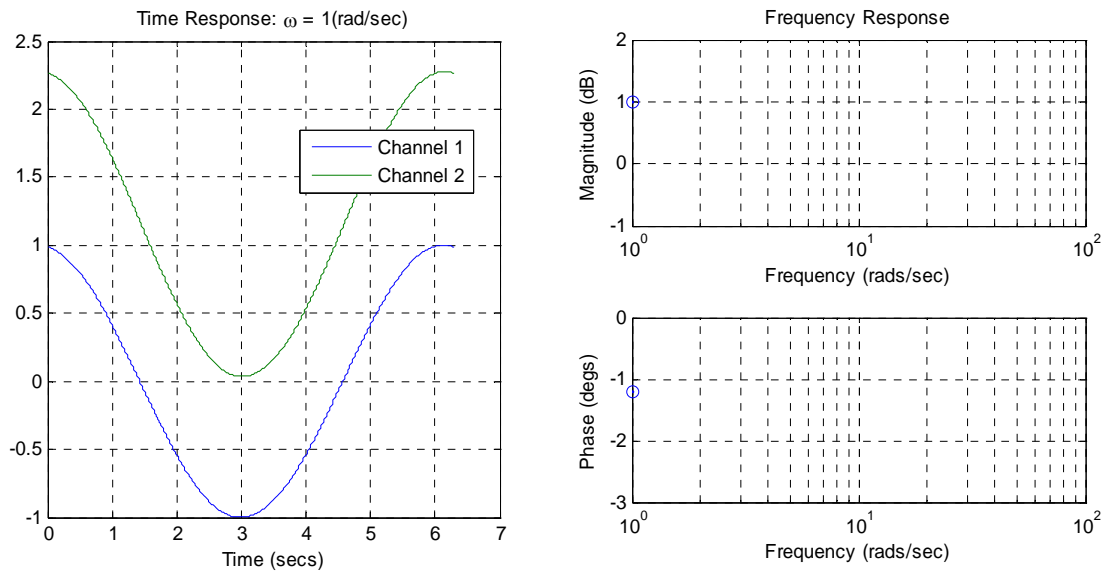


Figure 3: Dynamic signal analyzer output at $\omega = 1$ rad/sec .

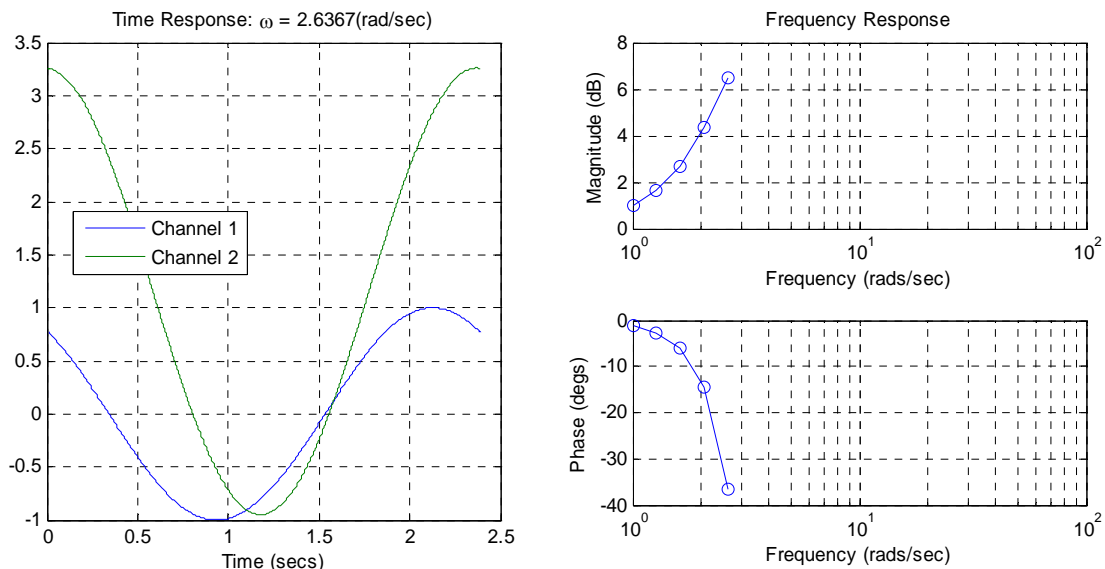


Figure 4: Dynamic signal analyzer output at $\omega = 2.6$ rad/sec .

much rougher. The ratio of the vehicle height amplitude to the roadway height amplitude is almost 7 dB. Furthermore, the vehicle is rising and falling after the roadway. This is indicated by the phase shift of almost -40 degrees. Furthermore, it can be seen from Figure 4 that due to the large amplitudes and phase shift, that the height of the vehicle is actually below the height of the roadway. This is physically impossible. More importantly, it implies that the suspension

system needs to be redesigned. Belanger corrects this problem by designing an active suspension system.[3] Note that this insight is not immediately obvious from the frequency response alone.

A comparison of the theoretical and the final “measured” frequency response are shown in Figure 5. The responses are nearly identical. If necessary or desired, the frequency response

could be measured at additional frequencies. As the students watched the measurements being taken after $\omega = 2.6$ rad/sec, they would see that magnitude of the vehicle height oscillations decrease and the phase shift. This should help to give them some intuition as to what the final frequency response means.

Dynamic Signal Analyzer Use in the Laboratory

While the DSA can provide the students in a classroom or a computer laboratory a significant amount of insight, it can provide them even more insight when combined with physical apparatus in the laboratory. A properly designed laboratory increases the students' "buy-in" to the concept, when they see it can actually be applied to a "real" system.

There are several different avenues for interfacing SIMULINK to the world outside the computer. Because of the ease of implementation, flexibility of approach, and the numerous experiments designed to work with the equipment, a platform from Quanser, Inc. was selected. A diagram of the experimental environment is shown in Figure 6. The experimental environment incorporates physical plants, analog and digital sensors, A/D and D/A converters, and the WinCon real-time interface with SIMULINK.

In this example, the frequency response of a direct current (DC) motor is measured. The frequency response could be used for system identification or control system design. This Simulink diagram, shown in Figure 7, is very similar to the one used for simulation in Figure 2. The suspension subsystem model has been

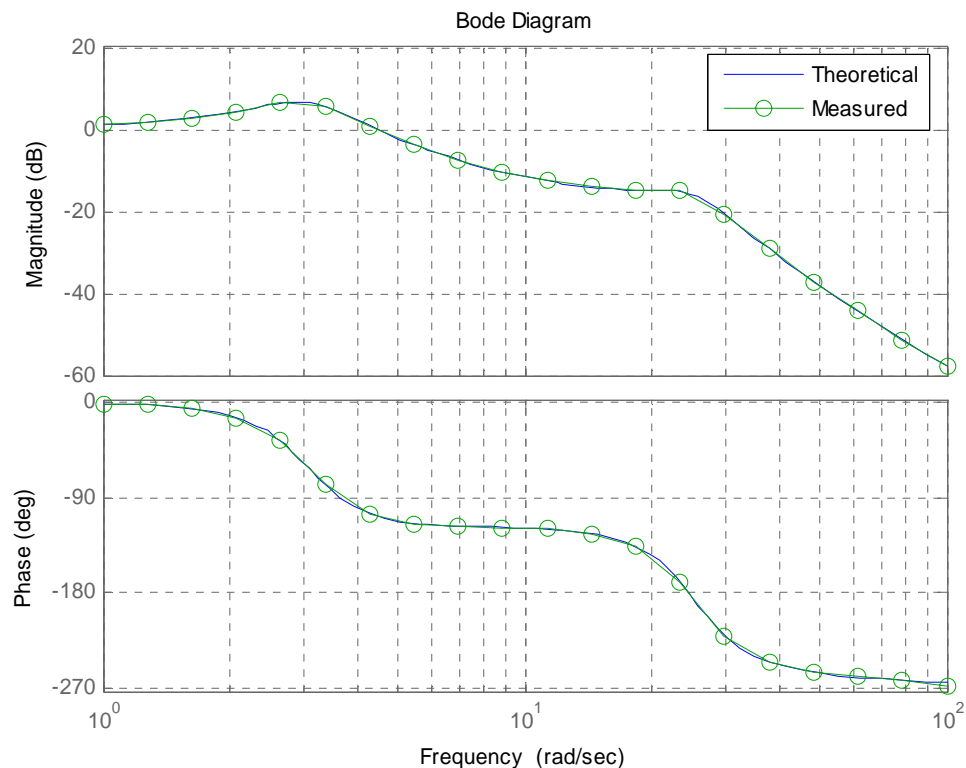


Figure 5: Comparison of theoretical and measured frequency response.

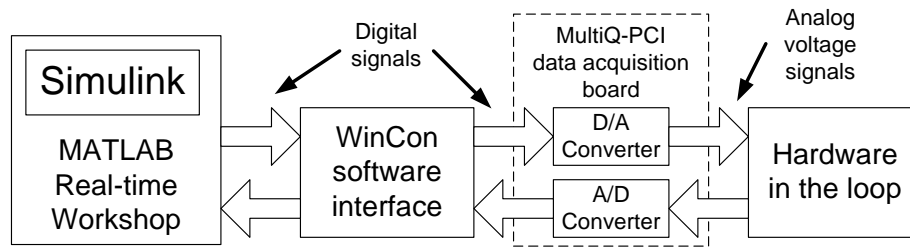


Figure 6: Experimental environment.

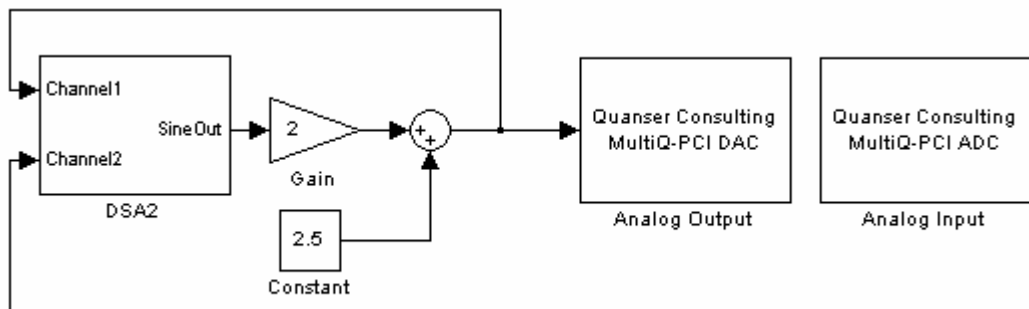


Figure 7: Simulink model with analog outputs and inputs with dynamic signal analyzer.

replaced with Analog Output and Analog Input blocks. The Analog Output block is connected to a power amplifier through a D/A converter on the Quanser input/output board. The power amplifier is used to drive the motor. The Analog Input block is connected to a tachometer through an A/D converter on the Quanser input/output board. The tachometer is used to measure the angular velocity and has units of 1 Volt/(rad/sec).

The output of the DSA block in Figure 7 is a unit amplitude sinusoid oscillating at a frequency specified by the MATLAB function developed for this block. In Figure 7, this signal has been amplified by 2.0 and offset by 2.5 producing a signal with a peak at 4.5 volts and a valley at 0.5 volts. This prevents the motor from stopping during the experiment. As a consequence, because the Coulomb friction enters the system as a DC disturbance, it has no effect on the measured frequency response. Furthermore, by eliminating the numerous starts and stops for high frequencies inputs, it serves to prolong the life of the motor. For more discussion of the laboratory setup and

the effect of Coulomb friction on the response, see the work by Watkins and O'Brien[4] and Esposito, Feemster, and Watkins.[5]

Once the Simulink model is developed and the hardware is connected, the method for taking the actual frequency response is almost identical to one used in simulation. The only significant difference is the fact that because we are working in real-time, the Simulink diagram must be compiled and linked before it can be run. However, this can be done with the click of one button. The frequency response of the DC motor is shown in Figure 8. From the frequency response shown in Figure 8, the students can easily identify the parameters of a model of the form

$$G(s) = \frac{\Omega(s)}{E_a(s)} = \frac{K_{mot}}{\tau s + 1} \quad (2)$$

where Ω is the angular velocity, E_a is the motor armature voltage, K_{mot} is the DC gain and τ is

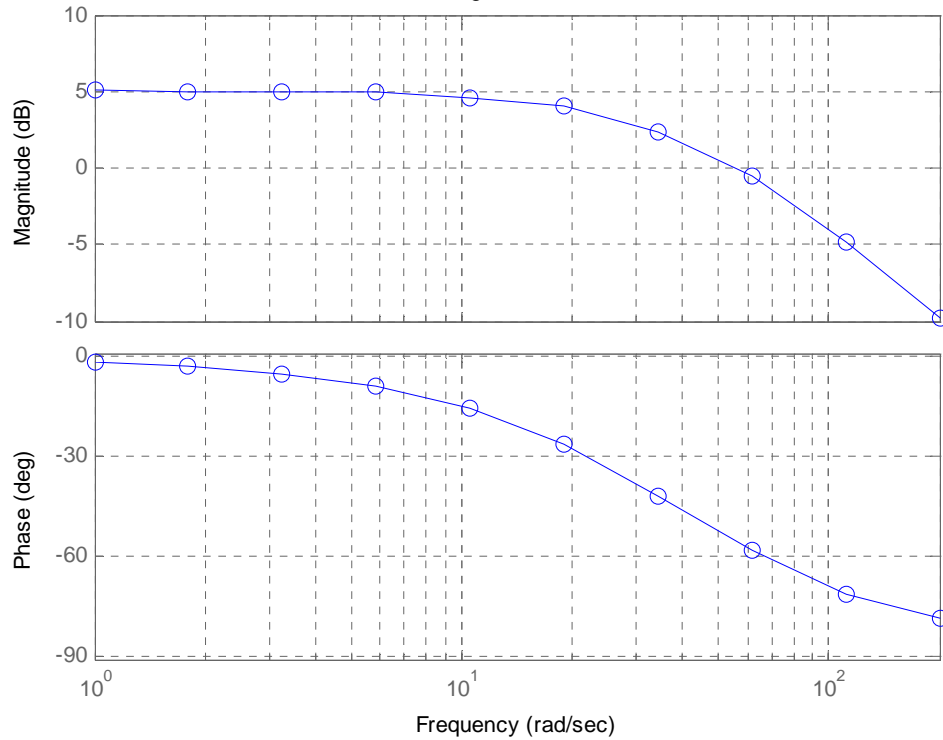


Figure 8: Frequency response of DC motor

the motor time constant. The frequency response can also be used directly for control system design. Because the data can be stored in the FRD model format, the frequency response plot can be easily plotted again using the MATLAB bode command. A Graphical User Interface (GUI) developed for the unified compensator design procedure by O'Brien and Watkins can easily be utilized[6].

Inside the Dynamic Signal Analyzer

In this last section, we will look inside the dynamic signal analyzer shown in Figure 9 to investigate how it works. The Simulink block is called by a MATLAB function at each frequency that it requests a measurement. The SineGen block outputs a sine wave at this frequency. To keep the analysis simple, we will assume that the sine wave may be amplified by a gain A , but that it has no offset. Consequently, the signal fed back into Channel 1 has the form

$$u(t) = A \sin(\omega t + \phi) \quad (3)$$

Also to keep the analysis simple, we will assume that the system, whether in real-life or in simulation, is linear and has a transfer function $G(s)$. Consequently, the output of the linear system that is fed into Channel 2, has the following form after the transients have settled

$$y(t) = A |G(j\omega)| \sin(\omega t + \phi + \angle G(j\omega)) \quad (4)$$

Note that the DSA waits until the transients have settled to take the measurements. The goal of the DSA is to find $G(j\omega)$.

It is not hard to show, that if we multiply (3) by $(\sin(\omega t) + j \cos(\omega t))$ and integrate over an integer number of periods that we get the following result

$$X = A \cos(\phi) + jA \sin(\phi) = A e^{j\phi} \quad (5)$$

The resulting measurement of $G(j\omega)$ as well as the time data is sent back to the MATLAB function where they are both plotted. While this analysis only looks at the ideal case, the actual performance achieved is often similar. This is due to the fact that high signal to noise ratios are achieved because we are only looking at signals at the frequency of interest.

Conclusions

The ability to understand and utilize the frequency response of a linear system is a critical building block in many undergraduate engineering disciplines. Yet despite significant exposure to and need for the frequency response, many undergraduate students have limited intuition on what it means and represents. Furthermore, equipment for measuring the frequency response is often too expensive and complex for a typical undergraduate engineering lab. Consequently, we have developed a virtual dynamic signal analyzer in SIMULINK for measuring the frequency response. To run the virtual DSA in simulation mode requires only SIMULINK. However, if combined with the Quanser WinCon software and hardware input/output board, it can be used for measuring the frequency response of experimental apparatus in the laboratory.

In this paper, it has been shown how this DSA can be used in the classroom and in the laboratory. A short discussion of how the DSA works has also been included. The virtual dynamic signal analyzer is available from the author upon request. A graphical user interface is being developed to make the DSA even easier to use.

References

1. Wang, F., Abramovitch, D., and Franklin, G., "A Method for Verifying Measurements and Models of Linear and Nonlinear Systems," *Proceedings of the American Controls Conference*, San Francisco, CA, June 1993.

2. Lilienkamp, K. and Trumper, D., "Dynamic Signal Analyzer for dSPACE," *Proceedings of the dSPACE User's Conference*, Dearborn, MI, May 2000.
3. Belanger, P., *Control Engineering: A Modern Approach*, Saunders College Publishing, Orlando, Florida, 1995.
4. Watkins, J. and O'Brien, R., "A Novel Approach to a Control Systems Laboratory," *Proceedings of the 2003 ASME International Mechanical Engineering Congress & Exposition*, Washington, DC, November 2003.
5. Esposito, J., Feemster, M. and Watkins, J., "Role of a MATLAB Real-Time Hardware Interface Within a Systems Modeling Course," *Computers in Education Journal*, Vol. 16, No. 1, pp. 41-50, 2006.
6. Watkins, J. and O'Brien, R., "A Unified Approach For Teaching Root Locus And Bode Compensator Design," *Proceedings of the American Control Conference*, Denver, CO, June 2003.

Biographical Information

John M. Watkins is an Associate Professor in the Department of Electrical and Computer Engineering at Wichita State University. He received his B.S. degree in electrical engineering from the University of Nebraska-Lincoln in 1989 and his M.S. and Ph.D. degrees in electrical engineering from The Ohio State University in 1991 and 1995, respectively. His research interests include feedback control systems, spacecraft dynamics and control, and networked control systems.