# The Use of Animation for Visualization of Concepts In a Network Analysis Class

Robert Bernick
California State Polytechnic University, Pomona

## Abstract

Many new and challenging concepts are introduced in the typical junior level network analysis class including s-plane analysis, convolution, and Fourier transforms. This paper discusses a set of animated clips that has been developed using Mathcad PLUS to aid in the visualization of these concepts in a way not possible with conventional lectures and textbook presentations. The clips are generated using the Mathcad FRAME variable. This variable takes on integral values that increase sequentially with time. Other variables can be defined that depend on the FRAME variable, and these can be plotted to provide a series of frames which become an animation clip when viewed sequentially. When the clip is saved, it is converted to a Windows avi file and can subsequently be played on any computer possessing the Windows operating system (version 3.1 or later).

Examples of the clips include sequences that show the relation of frequency response for second order lowpass and bandpass filters to the location of poles in the s-plane, the time evolution of the convolution integral, and the connection between the time duration of a signal and the frequency response of its Fourier transform.

Although the animation files are each around 500 KB in length, they are highly compressible and can typically be zipped to 10-15 KB. A set of animated clips has been made available to students and response has been positive.

## 1. **Introduction**

In the typical junior level networks class (continuous-time linear circuits) students are faced with a variety of new abstract concepts and methods of analysis. Among these are s-plane analysis, Fourier series and transforms, and convolution. While learning the required mathematical techniques may be straightforward, a deeper intuitive level of understanding is more difficult to achieve. Illustrations of concepts on blackboards and in textbooks are somewhat limited in their usefulness since they are static in nature. Animated pictures add a dynamic aspect, that can greatly enhance the visualization of abstract concepts. Animation as an educational tool in engineering has been carried out using MATLAB[1-4], Visisim[1] and Mathcad[5]. In addition web sites have been developed that employ Java applets in interactive displays which include animation[6, 7]. For the present application the capabilities of Mathcad PLUS 6.0 have proven ideal for developing a set of animated clips. The Mathcad script is intuitive, simple to use and does not require any high level programming language. After they are completed, the clips are compiled into animation files with the .avi extension. They can be played on any computer containing the Windows operating system (version 3.1 or later). Thus playing the animation clips does not require Mathcad. The student viewing the clips can control the speed of the animation, stop it at any point, repeat any segment and even run it backwards if desired by simply using a mouse.

Currently seven clips have been developed with others in progress. Topics covered by them include:

1. the relation between the location of poles in the s-plane and frequency response for biquadratic lowpass and bandpass filters
2. the time evolution of the convolution integral
3. the effect of the number of terms in a truncated Fourier series on the effectiveness of the series approximation
4. the relation between the time duration of a signal and the frequency spectrum of its Fourier transform.

## 2. **Mathcad Script**

In this section the Mathcad script, including the FRAME variable used to produce the animated clips, is discussed. The Mathcad FRAME variable takes on integral values that increase with time at a predetermined rate. Its usage is described in detail in the Mathcad User's Guide[8]. For every value of this variable, one animation frame is created. Other independent variables can be defined in terms of the FRAME variable so that their values also change from frame to frame. Examples of such variables used for the animations described in this paper are time, distance of a pole from the real axis, number of terms in a truncated Fourier series and the time duration of a pulse. Other variables can be plotted which depend on the FRAME-related independent variable. The result is a series of snapshot plots, each corresponding to one FRAME value. The animation is created when the snapshots are viewed sequentially.

One example of Mathcad script used to generate an animated sequence is shown in Figure 1. This example is designed to show how the gain of a biquad lowpass filter changes as the poles move toward the imaginary axis in the s-plane. The initial positions of the complex-conjugate poles are at $-1 \pm j$. The imaginary parts of the poles are maintained at $\pm j$ while the real part, which depends on the FRAME variable, is decreased in steps to 0.05. The filter gain in dB is plotted for each pole position.

$$\omega := .1, .2 .. 10$$

$$a := 1 - \frac{FRAME}{20}$$

$$real := -a$$

$$b(a) := 1$$

$$T(\omega) := \frac{1}{\sqrt{\left(1 - \frac{\omega^2}{a^2 + b(a)^2}\right)^2 + \left(2 \cdot a \cdot \frac{\omega}{a^2 + b(a)^2}\right)^2}}$$

$$A(w) := 20 \cdot \log(T(\omega))$$

$$Q := \frac{\sqrt{a^2 + b(a)^2}}{2 \cdot a}$$

Figure 1. Mathcad script for moving pole example

As a second example consider the script shown in Figure 2. Here the quantity tau represents a variable that depends on FRAME and which represents the half-width of a pulse in time f(t) which is centered at t=0. As tau increases the pulse widens. F(ω) represents the Fourier transform of the pulse. In the animation, both f(t) and F(ω) are plotted for each value of tau.

$$tau := \frac{FRAME}{10} + .2$$

$$t := -tau, -tau + .01 .. tau$$

$$f(t) := if(-tau < t < tau, 2, 0)$$

$$\omega := -20, -19.99 .. 20$$

$$F(\omega) := \left| \frac{4 \cdot \sin(\omega \cdot tau)}{\omega} \right|$$

$$Pulsewidth := 2 \cdot tau$$

Figure 2. Mathcad Script for Fourier transform example

## 3. **The Clips**

Each clip consists of one .avi file, typically about 500 KB long. Fortunately, these clips are highly compressible and when zipped only occupy between 10 and 15 KB of memory. The full set of animated clips is available for download in a zipped file at the author's website: http://www.intranet.csupomona.edu/~rlbernick. Here snapshots of the two examples discussed in section 2 obtained using the Windows 98 Media Player are shown for specific values of the FRAME variable. Figure 3 shows such a snapshot for the animated moving pole sequence described in section 2. In the snapshot the plot on the left shows the second quadrant pole located at the point -0.5 + j in the s-plane. The corresponding filter gain is shown to the right. At the bottom of the figure are shown the current values of the pole real part and the filter Q. In the animated clip the pole moves toward the imaginary axis. As it does so the gain curve peak rises and becomes sharper reflecting the increasing circuit Q. Using a mouse to control the slider at the bottom of the screen the pole can be moved back and forth with corresponding changes in the gain curve and the Q. Watching the clip gives the observer a "feel" for the significance that the filter pole position has on frequency response.

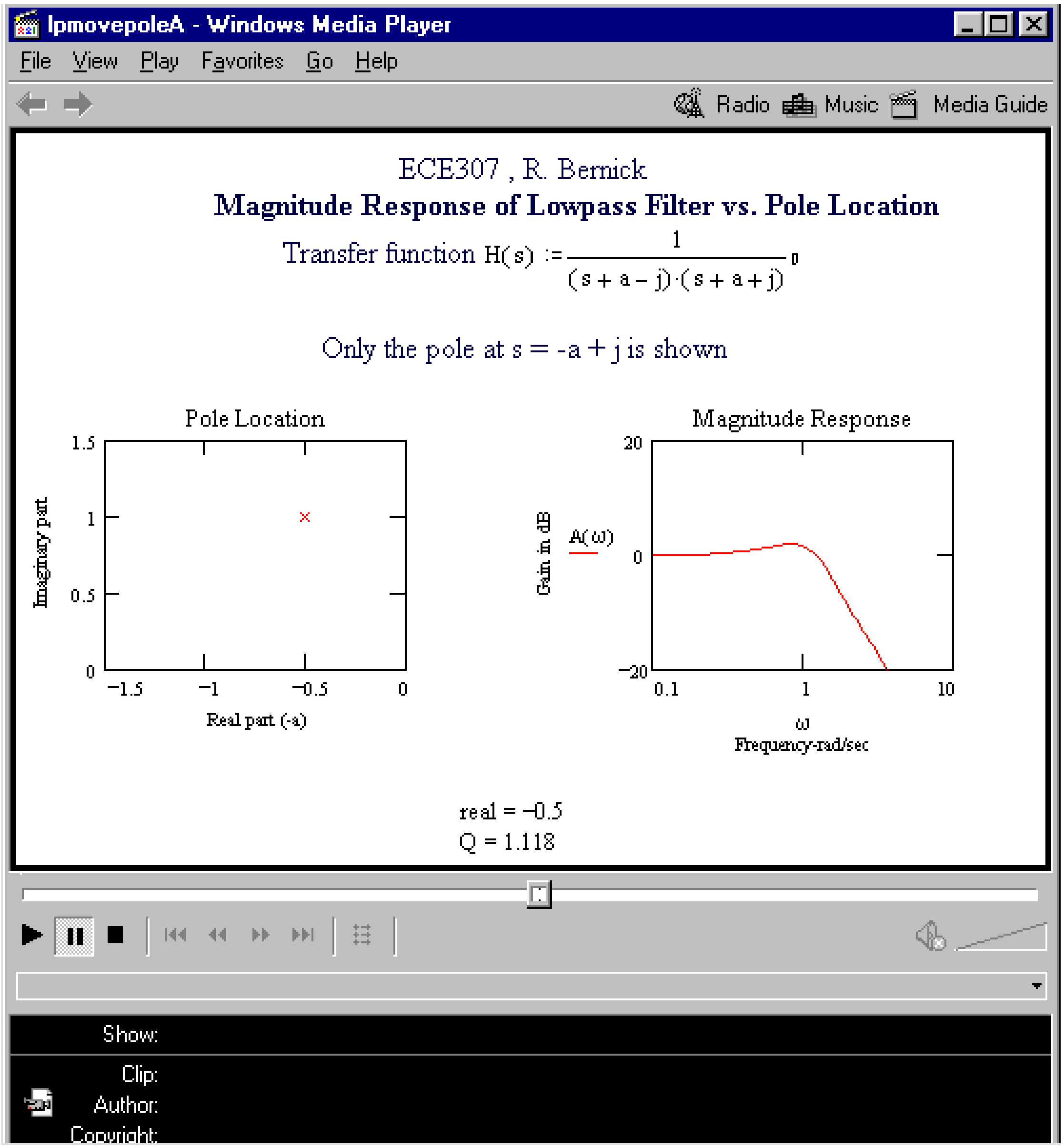


Figure 3. Snapshot of moving pole animation example

Figure 4 is a snapshot related to the second example discussed in the section 2. Here a pulse with a width of 2 seconds is shown in the plot on the left. The plot on the right shows the corresponding amplitude spectrum of the pulse's Fourier transform. In the animated sequence as the pulse width increases, the Fourier transform becomes narrower and the peak value increases. Again the viewer can manipulate the slider under the plots to vary the pulse width as desired while observing the resulting transform graph. Thus the clip provides a visualization of the relationship between the time duration of a signal and its frequency spectrum.

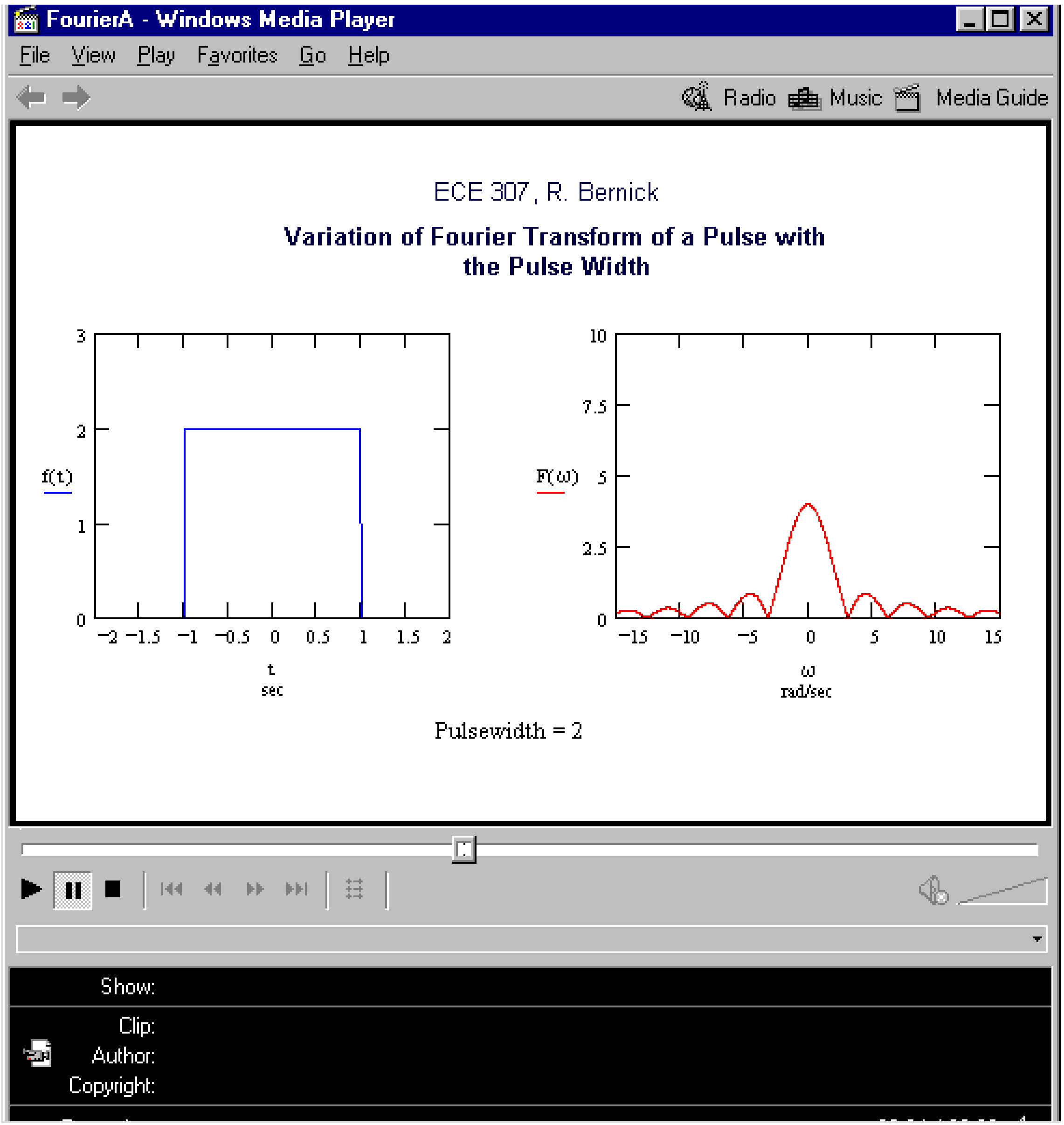


Figure 4. Snapshot of Fourier transform animation example

The other clips that have been developed have a similar format utilizing two plots that can be viewed simultaneously.

## 4. **Conclusion**

A simple means of using Mathcad to generate animated clips for use in a junior level network analysis class has been described. The clips are intended to enhance student understanding of abstract ideas introduced in this class by providing a dynamic means of visualizing concepts. Examples of the animation generating Mathcad script have been discussed and snapshots of the resulting animation shown. The clips have been made available to students taking the class at the author's institution and their reaction has been positive. A zipped file containing the complete set of clips is available for download at the author's website.

## Bibliography


1. Watkins, J., Piper G., Wedeward, K., and Mitchell, E., "Computer Animations: A Visualization Tool for Dynamic Systems Simulations", *Proceedings of the 1997 ASEE Annual Conference,* session 1620, June 1997.
2. Jenkins, B., "Simulation and Animation in Optical Fiber Communication", *Proceedings of the 1998 ASEE Annual Conference,* session 1620, June 1998.
3. Brooking, C., and Smith, D., "Simulation and Animation of Kinematic and Dynamic Machinery Systems with MATLAB", *Proceedings of the 1998 ASEE Annual Conference,* session 1620, June 1998.
4. Mechtel, D., Field, C., and Jenkins, B. "Graphical Analysis and Animation in an Introductory Electrodynamics Course", ", *Proceedings of the 1999 ASEE Annual Conference,* session 2520, June 1999.
5. Aburdene, A., Zarrouk, R., and Magargle, R., "Interactive Tutorial for an Introductory Electrical Engineering Course", *Proceedings of the 2000 ASEE Annual Conference,* session 2793, June 2000.
6. URL: http://jas2.eng.buffalo.edu/applets/index.html, "Semiconductor Applets", SUNY, Buffalo.
7. URL: http://www.jhu.edu/~signals/, "Signals, Systems, Control Demonstrations", Johns Hopkins University.
8. <u>Mathcad User's Guide</u>, Mathsoft, Inc., Cambridge, Massachusetts, 1995.



ROBERT BERNICK is a Professor of Electrical and Computer Engineering at the California State Polytechnic University, Pomona. He previously worked in the aerospace industry. He received the BA degree in mathematics from the University if Minnesota, the MA degree in physics from the University of California, Berkeley, and the Ph.D. degree in physics from the University of Southern California.