AC 2011-2172: RECENT ADVANCES IN COMPUTATIONAL TECHNOLOGY IN THE CLASSROOM

Mariusz Jankowski, University of Southern Maine

Dr. Mariusz Jankowski received the Ph.D. in Electrical Engineering from the City University of New York in 1989. He is currently an Associate Professor of Electrical Engineering and Chairperson at the University of Southern Maine. His research interests are in the areas of signal and image processing, in particular image enhancement, segmentation, shape description, and recognition. He has many years of experience in developing commercial software for image processing and is the author of a professional software system for image processing based on Mathematica, a modern system for scientific computing. Dr. Jankowski has received awards from the Ames Laboratory, Wolfram Research, and University of Southern Maine for his scholarly and pedagogic work. He has received grants from the National Science Foundation, Maine Science and Technology Foundation, and Wolfram Research.

Recent advances in computational technology in the classroom

Abstract

Recent advances in computational technology have made it significantly easier to create interactive demonstrations with programmable tools that are fully integrated within a leading mathematical software system. This paper describes how the new dynamic interactivity language in addition to standard features built into *Mathematica* are being used in teaching selected sophomore and senior undergraduate electrical engineering classes at the University of Southern Maine. It describes how several typical problems encountered in integrating advanced computational systems into an undergraduate curriculum have been addressed. Particular attention will be paid to the creation and classroom use of demonstrations illustrating some core ideas such as convolution, filtering, and frequency response.

Introduction

Recent advances in software technology in *Mathematica*, a leading mathematical software system from Wolfram Research (www.wolfram.com), have made it significantly easier to create interactive calculations for exploring the solution space of a problem or for classroom demonstration purposes. The addition of a comprehensive set of control objects and a dynamic interactivity language consisting of a few powerful new constructs that automatically track dependencies between control objects and symbols has resulted in a very simple mechanisms for creating interactive content. This paper describes how *Mathematica* is being used in teaching selected undergraduate electrical engineering classes at the University of Southern Maine (USM) and shows how many of the typical problems encountered in integrating advanced computational systems into an undergraduate curriculum have been addressed by the system's features and functionality. Particular attention will be paid to the creation and classroom use of interactive content demonstrating how it is used in explaining some core ideas such as convolution, filtering, and frequency response. The integration of computation into a curriculum is as old as the 1970s, when it was proposed as a means of improving undergraduate mathematics education. It is now widely accepted within the engineering disciplines as a necessary and beneficial component of the curriculum[1, 2]. As a result, practically all recent textbooks in areas of electrical engineering such as circuits, signals and systems, digital signal processing, or control include computational modules, usually based on the popular Matlab (www.mathworks.com) system. Such courses typically require significant algebraic skills in manipulating a variety of mathematical expressions. In fact, recently published data gathered in the Signals and Systems Concepts Inventory project[3] confirms that a student's mathematical skills are a dominant factor in predicting the student's success in learning key concepts of signals and systems. Conversely, the inability to perform these manipulations may be a significant barrier. As a result, signal and systems textbooks devote whole sections and appendices to such topics as finding solutions to systems of ordinary constant-coefficient difference or differential equations, partial fraction expansion, the evaluation of many types of integrals and sums, and complex numbers and functions. These are precisely the type of computations that can be accomplished easily with *Mathematica* using familiar mathematical notation. State-of-the-art computer algebra is widely recognized to be one of the premier strength of *Mathematica* and is one of the reasons why it was selected as the tool to teach selected electrical engineering courses at USM. As an example consider the evaluation of a convolution integral, specifically, the response of a first-order RC filter to a pulse input[4]. This defines the input and impulse response signals as piecewise functions on the domain $-\infty < t < \infty$.

```
h[t_] := Piecewise[{{ⅇ^-t, t ≥ 0}}] ;
x[t_] := Piecewise[{{1, 0 ≤ t ≤ 2}}];
```

This shows the calculation and returns the result. Significantly, note the use of standard mathematical notation and that integration over piecewise functions is fully supported.

$$\int_{-\infty}^{\infty} h[\tau]\, x[t-\tau]\, ⅆ\tau$$

$$\begin{cases} e^{-t}\left(-1+e^{2}\right) & t > 2 \\ e^{-t}\left(-1+e^{t}\right) & 0 < t \le 2 \\ 0 & \text{True} \end{cases}$$

Alternatively, the same result can be obtained using the built-in function `Convolve`.

```
Convolve[x[τ], h[τ], τ, t]
```

$$\begin{cases} e^{-t}\left(-1+e^{2}\right) & t>2 \\ 1-e^{-t} & 0<t\le 2 \\ 0 & \text{True} \end{cases}$$

The result can be easily plotted (see Figure 1) by referencing the most recent output using the symbol %.

```
Plot[%, {t, -2, 10}]
```

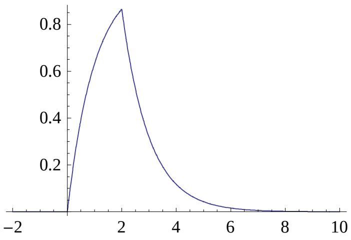


Figure 1. Plot of impulse response of first-order RC filter using *Mathematica* function `Convolve[]`.

Text, graphics, and computation

As demonstrated in the earlier example, *Mathematica* is a general purpose computational system that successfully combines the three components of text, graphics, and computation. (In fact, this paper was written entirely in *Mathematica*.) The combination of computation and graphics helps in demonstrating import concepts by means of visualization. The addition of explanatory text makes it a convenient system for creating technical documents such as lectures, presentations, reports and more. Consider the following example of several lines of calculations and visualizations showing the transfer function and frequency response of a 5th-order Butterworth filter system. This defines the transfer function.

$$H[s\_] := \frac{1.}{\left(1.0+3.23607\,s+5.23607\,s^{2}+5.23607\,s^{3}+3.23607\,s^{4}+s^{5}\right)}$$

Figure 2 shows a surface plot of $|H(s)|$, the magnitude of the system transfer function with the magnitude evaluated on the complex plane defined by the rectangular coordinates σ and ω, and the symbol i represents the imaginary number $\sqrt{-1}$.

```
Plot3D[Abs[H[σ + ⅈ ω]], {σ, -3, 1}, {ω, -2, 2}]
```

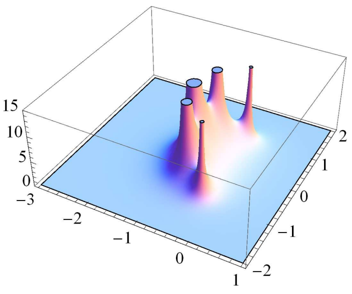


Figure 2. Surface plot of the magnitude of the transfer function of a 5th order Butterworth filter.

Here is the magnitude of the frequency response $|H(j\,\omega)|$ (Figure 3). Note that the frequency response is obtained by evaluating the system transfer function $H(s)$ on the imaginary axis.

```
Plot[Abs[H[ⅈ ω]], {ω, -π, π}]
```

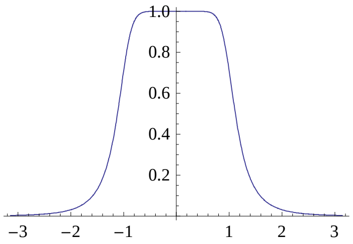


Figure 3. Magnitude response of a 5th order Butterworth filter.

Finally, here is the so-called Bode plot of the magnitude and phase spectra of the system defined by $H(s)$ (Figure 4).

```
BodePlot[H[s], GridLines → Automatic, PlotLayout → List]
```

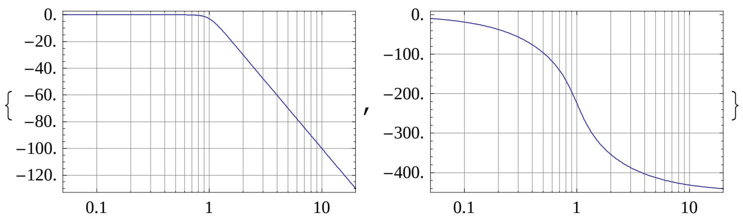


Figure 4. Bode plot a 5th order Butterworth filter.

However, these are just the minimal requirements. Overcoming the unfamiliarity of the syntax of any given computational language is a common obstacle to using code and computation to illustrate core concepts of a discipline. For this reason, standard languages such as C, C++, Java have not shown to be effective in teaching and learning circuits, signal and systems, signal processing, control and similarly mathematical areas of electrical engineering. It is much more convenient to use a system such as *Mathematica* which supports properly typeset, familiar mathematical notation. This can be demonstrated with two simple examples. First, the discrete-time Fourier transform (DTFT) of a causal, real, exponential sequence[5].

$$\sum_{n=0}^{\infty} a^n e^{-i \omega n}$$

$$\frac{e^{i\omega}}{-a + e^{i\omega}}$$

Next, the inverse DTFT of an ideal, full-band differentiator.

$$\int_{-\pi}^{\pi} i\, \omega\, e^{i \omega n}\, d\omega$$

$$\frac{2\, n\, \pi\, \mathrm{Cos}[n\, \pi] - 2\, \mathrm{Sin}[n\, \pi]}{n^2}$$

This simplifies the expression using a symbolic rule and a replace operation. (Symbol % indicates the most recent output.)

```
% /. Sin[n π] → 0
```

$$\frac{2\, \pi\, \mathrm{Cos}[n\, \pi]}{n}$$

As demonstrated, a *Mathematica* notebook is simultaneously a platform for computation and a technical document. These two forms complement themselves in a classroom setting allowing the instructor to freely mix two types of classroom activities. Notebooks, like standard textbooks, may include explanatory text, typeset mathematical equations and graphics, but unlike textbooks also include computable mathematical expressions and programs. In the signals and systems course, each notebook is written so as to maximize the opportunities for live computation during a lecture or laboratory session. The computational capabilities in *Mathematica* are used in a variety of ways. At its most basic level *Mathematica* may be used as a simple, but powerful calculator which students can use to solve many of the problems found in standard undergraduate textbooks. The student benefits by spending more time formulating and understanding the problem than on algebraic manipulation. Examples of algebraic evaluation common to many introductory undergraduate courses include:

- solutions to constant coefficient difference and differential equations,
- solutions of systems of (non)linear equations,
- evaluation of convolution sums and integrals,
- evaluation of Fourier, Laplace and z-transform integrals and sums,
- partial fraction expansion of rational polynomials.

Here is a solution to a system of linear equations using the function `Solve`, as would be typical of a course on network analysis[6]. The result gives the values of the node voltages v_1, v_2, and v_3 in the given circuit.

$$\texttt{Solve}\left[\left\{\frac{v_1 - v_3}{6\,k} + \frac{v_1}{12\,k} + \frac{v_2}{6\,k} + \frac{v_2 - v_3}{12\,k} == 0,\ v_1 - v_2 == 2\,v_2,\ v_3 == 6\right\},\ \{v_1, v_2, v_3\}\right]$$

$$\left\{\left\{v_1 \to \frac{9}{2},\ v_2 \to \frac{3}{2},\ v_3 \to 6\right\}\right\}$$

This returns the unit impulse response of a second-order discrete-time system[7] using the difference equation solver `RSolve`.

$$\begin{aligned}&\texttt{RSolve}\left[\left\{\texttt{y[n]} - \frac{5}{6}\,\texttt{y[n-1]} + \frac{1}{6}\,\texttt{y[n-2]} == \texttt{KroneckerDelta[n]},\right.\right.\\ &\qquad \left.\left.\texttt{y[-1]} == 0,\ \texttt{y[-2]} == 0\right\},\ \texttt{y[n]},\ \texttt{n}\right]\ \texttt{// First}\end{aligned}$$

$$\left\{\texttt{y[n]} \to 6^{-n}\left(2^{n}\left(\begin{cases}-2 & n > -2\\ 0 & \text{True}\end{cases}\right) + 3^{n}\left(\begin{cases}3 & n > -2\\ 0 & \text{True}\end{cases}\right)\right)\right\}$$

This shows the result in a simpler form.

$$\texttt{PiecewiseExpand[y[n] /. \%]}$$

$$\begin{cases}-6^{-n}\left(2^{1+n} - 3^{1+n}\right) & n > -2\\ 0 & \text{True}\end{cases}$$

This evaluates the response for a range of values of $n = -2, -1, \ldots, 10$.

$$\texttt{Table[\{n, \%\}, \{n, -2, 10\}]}$$

$$\begin{aligned}&\left\{\{-2, 0\}, \{-1, 0\}, \{0, 1\}, \left\{1, \frac{5}{6}\right\}, \left\{2, \frac{19}{36}\right\}, \left\{3, \frac{65}{216}\right\}, \left\{4, \frac{211}{1296}\right\}, \left\{5, \frac{665}{7776}\right\},\right.\\ &\left.\left\{6, \frac{2059}{46\,656}\right\}, \left\{7, \frac{6305}{279\,936}\right\}, \left\{8, \frac{19\,171}{1\,679\,616}\right\}, \left\{9, \frac{58\,025}{10\,077\,696}\right\}, \left\{10, \frac{175\,099}{60\,466\,176}\right\}\right\}\end{aligned}$$

If preferred, the numerical result can be obtained directly by using the numerical difference equation solver `RecurrenceTable`.

$$\begin{aligned}&\texttt{RecurrenceTable}\left[\left\{\texttt{y[n]} - \frac{5}{6}\,\texttt{y[n-1]} + \frac{1}{6}\,\texttt{y[n-2]} == \texttt{KroneckerDelta[n]},\right.\right.\\ &\qquad \left.\left.\texttt{y[-1]} == 0,\ \texttt{y[-2]} == 0\right\},\ \texttt{y[n]},\ \{\texttt{n}, 10\}\right]\end{aligned}$$

$$\begin{aligned}&\left\{0, 0, 1, \frac{5}{6}, \frac{19}{36}, \frac{65}{216}, \frac{211}{1296}, \frac{665}{7776},\right.\\ &\left.\frac{2059}{46\,656}, \frac{6305}{279\,936}, \frac{19\,171}{1\,679\,616}, \frac{58\,025}{10\,077\,696}, \frac{175\,099}{60\,466\,176}\right\}\end{aligned}$$

Mathematica as a technical documentation platform for presentation and computation has been discussed highlighting its use in an undergraduate electrical engineering program. It is important to add that while all the features presented to this point played an important role in selecting it as the preferred computational system for the program, it is the unique combination of outstanding symbolic and numerical evaluation capabilities that were ultimately the most important. In its early days, *Mathematica* was known as a leading computer algebra system and it is probably fair to say that it remains so to this day. It is the symbolic capabilities that give a user closed form solutions to difference and differential equations, Fourier, Laplace, z - transforms and more. These are exactly the sort of calculations that a typical course on circuits, signals and systems, or digital signal processing demands and which can be easily accomplished with *Mathematica* using familiar mathematical notation. Finally, *Mathematica*'s state-of-the-art numerical capabilities and its rich and efficient programming language need to be added to its list of features, all together resulting in an outstanding platform for everyday computing.

Dynamic content

The barriers in mastering any particular software system are typically considered the main reason why computation is not more widely used in teaching and learning. Even *Mathematica* despite its mostly mathematical notation, a proliferation of palettes to help users enter syntactically correct commands, syntax coloring, and extensive on-line help, is still considered by many as difficult to master. Therefore it comes as no surprise that the drive behind creating web-based instructional modules stems mainly from the desire to hide implementation and avoid the specifics of any one computer language. Other reasons for the proliferation include availability of tools for interactivity, creating animations, integration of sound and video, and lastly and importantly, universal availability. Interestingly, a recent version of *Mathematica* introduced a whole range of new functionality in the area of interactivity and graphical user interface creation. A comprehensive set of control objects and a dynamic interactivity language consisting of a few powerful new constructs that automatically track dependencies between control objects and symbols were added to the system. This has turned out to be particularly useful in a classroom setting. It allows the creation of live, interactive demonstrations that can be used to easily and naturally explore solution domains. Interactive demonstrations are a powerful abstraction tool that till now has been far too difficult for most of us to create quickly on an "as needed" basis. Finally, since these tools are fully integrated into the *Mathematica* system, the implementation behind any demonstration can be easily opened for inspection and discussion.

Consider again, the topic of convolution, one of those notoriously difficult concepts and calculations in signals and systems courses. The continuous-time case is especially difficult for today's students as it requires a multiplicity of mathematical skills that are often found lacking, including visualization of time-reversal, time shift, ability to integrate functions over multiple regions, the determination of correct formulations for functions on selected intervals and more. The addition of interactive content has the potential of clarifying some of these concepts.

Here is a demonstration of the concepts of signal reversal and shifting (Figure 5). Note how easy it is to create interactive content, simply by wrapping a `Manipulate` function across a standard plot function. *Mathematica* does all the rest, it binds variables (i.e., t) to control object (the slider at the top of the plot) eliminating in most cases any need of writing low-level callback functions, designing the layout, and other tedious tasks usually associated with graphical user interface design.

```
Manipulate[Plot[{x[τ], x[t - τ]}, {τ, -3, 8}], {{t, -0.5}, -1, 5}]
```

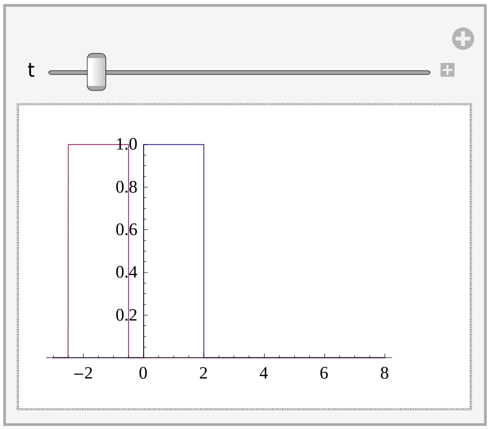


Figure 5. Snapshot of an interactive display showing an example signal with position of the signal dependent on the slider control object.

Here is a snapshot of an interactive plot (Figure 6) that shows the signal $x(t-\tau)$ sliding across the signal $h(\tau)$ and the area of intersection defined by the product $h(\tau)\,x(t-\tau)$ (as required by convolution).

```
Manipulate[Plot[{h[τ] x[t - τ], x[t - τ], h[τ]}, {τ, -3, 8}, Filling → {1 → Axis}],
 {{t, 1}, -1, 5}]
```

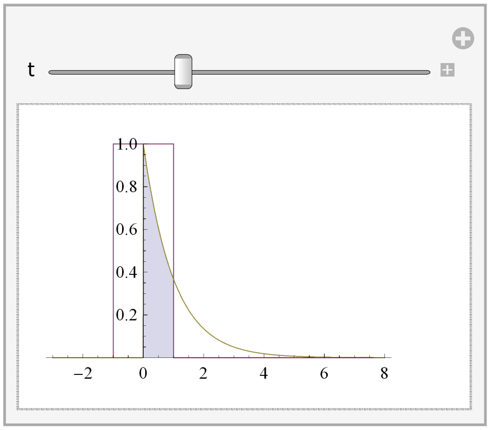


Figure 6. Snapshot of an interactive display showing signal area of overlap (shaded region) of signals $h(\tau)$ and $x(t-\tau)$ with value of t given by the slider control object.

What remains to be calculated and plotted is the area of the shaded region defined by the product $h(\tau)\,x(t-\tau)$. This, of course, is the result returned by the convolution integral evaluated in the Introduction section of this paper. Here it is shown again.

$$y[t\_] := \begin{cases} e^{-t}\left(-1+e^{2}\right) & t>2 \\ 1-e^{-t} & 0<t\leq 2 \\ 0 & \text{True} \end{cases};$$

A plot of the solution can be found in many textbooks on signals and systems, however, an interactive plot (Figure 7) has the advantage of clearly connecting time advance and the operation of sliding one signal with respect to the other, with the build-up of the result.

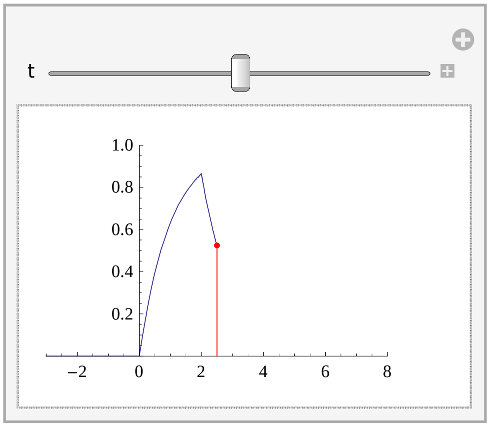


Figure 7. Snapshot of an interactive display showing the result of evaluating a convolution integral over the time interval $-\infty < \tau < t$ with value of t given by the slider control object.

Demonstrations

In the previous section we demonstrated how easily one can create interactive content to illustrate an idea or concept and embed it in a *Mathematica* notebook. However, it is sometimes desirable (or even necessary) to create a stand-alone or web-based applications. The demonstrations website at Wolfram Research (demonstrations.wolfram.com) currently holds over five thousand demonstrations uploaded by users, a testament to how relatively easy it is to create them.

In this section a demonstration based on a common laboratory exercise in introductory courses on circuits is created. The goal is to show how relatively easy it is to build simple yet instructive demonstrations. The demonstration highlighted in this section shows the response of a first-order, lowpass RC filter to selected periodic inputs, such as typically available on a laboratory waveform generator.

It complements a common laboratory exercise shown in Figure 8, in which students build the circuit, connect an oscilloscope to the input and output ports, and drive the circuit with a sinusoidal source using a function generator. This experiment serves several important purposes: to clearly demonstrate that sinusoidal inputs produce sinusoidal outputs in linear systems, to plot the so-called voltage gain, and therefore learn about the frequency response of this circuit, and to learn how the values of the resistor and capacitor impact the various properties of this circuit.

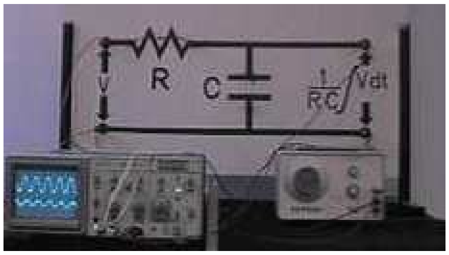


Figure 8. A simple RC-filter experiment. [Photo courtesy: http://www.physclips.unsw.edu.au/jw/RCfilters.html]

It is well-known that the output of the circuit discussed here can be calculated in a number of different ways including differential equations, phasors, and Laplace transform, to name the three most important approaches. Here we use the direct approach of solving for the output from the defining differential equation. The following command gives the general solution for the voltage across the capacitor *v*(*t*) for any frequency *ω*.

```
DSolve[{v'[t] + v[t] == Sin[ω t] UnitStep[t], v[0] == 0}, v[t], t] //
 First
```

$$\left\{\mathrm{v}[\mathrm{t}] \to -\frac{e^{-t}\left(-\omega + e^{t}\,\omega\,\mathrm{Cos}[t\,\omega] - e^{t}\,\mathrm{Sin}[t\,\omega]\right)\mathrm{UnitStep}[t]}{1+\omega^{2}}\right\}$$

Mathematica's differential equation solver is fast enough to allow for "real-time," meaning interactive evaluation for any input waveform of interest. We can therefore dispense with pre-calculating solutions to signals of interest and simply in-line the solver in the demonstration. For a sinusoidal source and assuming a frequency of $\omega = 1\,\frac{\text{rad}}{\text{sec}}$ we get the following graphical result, showing the input (in blue) and the resulting output (in red).

```
Plot[
 Evaluate[{Sin[ t] UnitStep[t],
   (v[t] /. First[DSolve[{v'[t] + v[t] == Sin[t] UnitStep[t], v[0] == 0},
       v[t], t]])}], {t, 0., 20.}, AxesLabel -> Automatic]
```

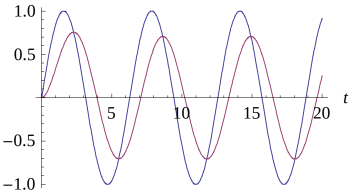


Figure 9. Plot of the input sinusoid (blue) and response (red) of a first-order RC-filter.

It is easy to set the frequency of the input signal as the interactive parameter of the demonstration, yielding the following graphical user interface with a slider as the control object (Figure 10).

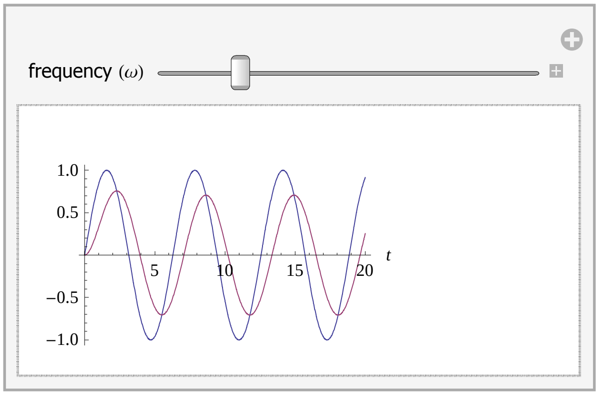


Figure 10. Snapshot of an interactive display showing the response of a first-order RC filter to a sinusoid with frequency given by slider control object.

With just a little bit more effort, controls for selecting different input waveforms can be added making it a more instructive and versatile demonstration. This demonstration is available at http://demonstrations.wolfram.com/ResponseOfLowPassRCFilterToPeriodicWaveforms/. Figure 11 shows three screen shots from the on-line demonstration.

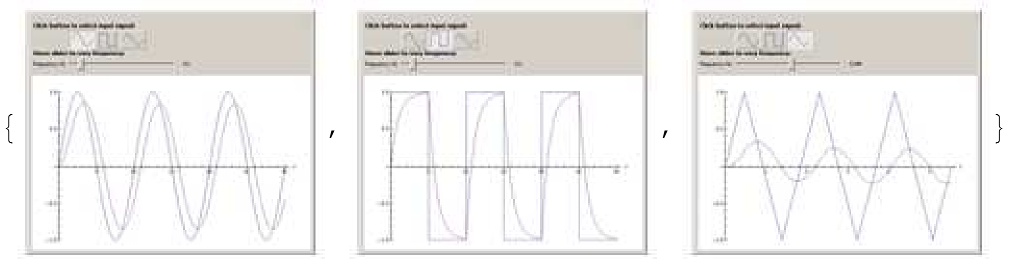
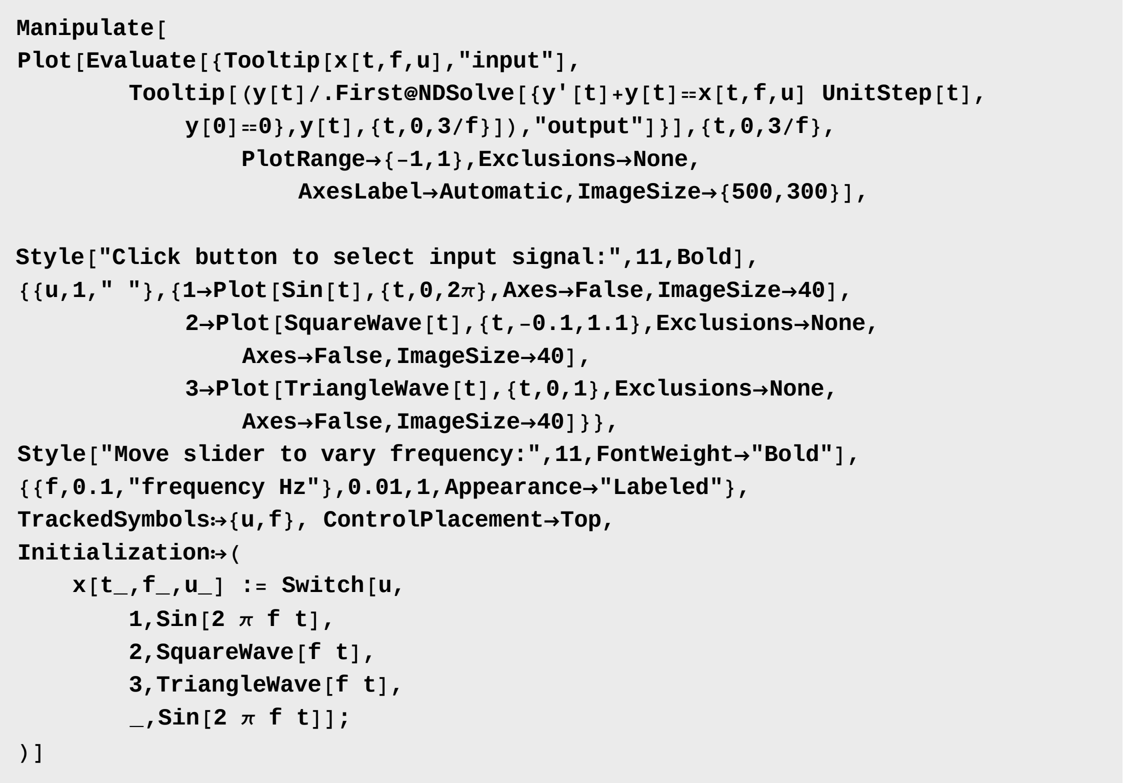


Figure 11. Screen shots of demonstration “Response of lowpass RC filter to periodic waveforms.”

It is interesting to consider the the full source code for the demonstration (see Figure 12). Note that the calculation neededcessary to display the result is in the line which invokes a `Plot` command immediately following the word `Manipulate`, with the output calculated using a differential equation solver inside the graphics function. The remainder of the code produces the buttons and slider control objects visible at the top of the display area and initializes the calculation. It can be said that the ease with which interactive demonstrations can be created in *Mathematica* compares favorably with any other system that is presently available.

Conclusion

This paper demonstrates key *Mathematica* features that make it almost uniquely suited for use in an undergraduate electrical engineering classroom. The features that play a particularly significant role include the notebook interface, the advanced computer algebra capabilities, and the dynamic interactivity language and control objects. We show how each of these features is used in creating instructional materials that support the teaching of a third-year Signals and Systems course at USM. We devote much of the paper to describing the most recently added feature allowing for easy creation of interactive input particularly useful in exploring solution spaces or classroom demonstrations. This is a significant enhancement of the already formidable system capabilities, adding a dydactically useful modality that previously required a significant effort and specialized expertise to develop and maintain. Each of the described features plays an important role in making *Mathematica* a noteworthy computational and pedagogic tool for the instructor and the student.



```
Manipulate[
Plot[Evaluate[{Tooltip[x[t,f,u],"input"],
        Tooltip[(y[t]/.First@NDSolve[{y'[t]+y[t]⩵x[t,f,u] UnitStep[t],
            y[0]⩵0},y[t],{t,0,3/f}]),"output"]}],{t,0,3/f},
                PlotRange→{-1,1},Exclusions→None,
                    AxesLabel→Automatic,ImageSize→{500,300}],

Style["Click button to select input signal:",11,Bold],
{{u,1," "},{1→Plot[Sin[t],{t,0,2π},Axes→False,ImageSize→40],
            2→Plot[SquareWave[t],{t,-0.1,1.1},Exclusions→None,
                Axes→False,ImageSize→40],
            3→Plot[TriangleWave[t],{t,0,1},Exclusions→None,
                Axes→False,ImageSize→40]}},
Style["Move slider to vary frequency:",11,FontWeight→"Bold"],
{{f,0.1,"frequency Hz"},0.01,1,Appearance→"Labeled"},
TrackedSymbols⧴{u,f}, ControlPlacement→Top,
Initialization⧴(
    x[t_,f_,u_] := Switch[u,
        1,Sin[2 π f t],
        2,SquareWave[f t],
        3,TriangleWave[f t],
        _,Sin[2 π f t]];
)]
```

Figure 12. Code for demonstration “Response of lowpass RC filter to periodic waveforms.”

References

[1] Schwingendorf, K.E. and E. Dubinsky. “Purdue University: Calculus, concepts, and computers: Innovations in learning.” In: Tucker, T.W. (Ed.), Priming the calculus pump: Innovations and resources, Washington, DC: The Mathematical Association of America, 1990. 175-198.

[2] Brown, D., H.A. Porta, and J.J. Uhl. “Calculus and *Mathematica*: A laboratory course for learning by doing.” In: L. C. Leinbach (Ed.), The laboratory approach to teaching calculus, Washington, DC: The Mathematical Association of America, 1991. 99-110.

[3] K.E. Wage, J.R. Buck, C.H. Wright, and T.B. Welch. “The Signals and Systems Concepts Inventory,” *IEEE Transactions on Education*, Vol. 48, No. 3, 2005. 448 - 460.

[4] Haykin, S. and B. V. Veen. “Chapter 2: Example 2.7.” Signals and Systems, John Wiley & Sons, 2005 (2nd Ed). 119-120.

[5] Oppenheim, A.v., A.S. Willsky, and S.H. Nawab. “Chapter5: Example 5.1.” Signals & Systems, Prentice Hall, 1996 (2nd Ed). 362-363.

[6] Irwin, J.D. and R.M. Nelms. “Chapter 3: Example 3.9.” Basic Engineering Circuit Analysis, John Wiley & Sons, 2008 (9th Ed). 112.

[7] McClellan, J.H., R.W. Schafer, and M.A. Yoder. “Chapter 8: Example 8.15.” Signal Processing First, Pearson/Prentice Hall, 2003. 227-228.